

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITECNICA SUPERIOR

INGENIERIA TECNICA INDUSTRIAL

MECANICA



PROYECTO FIN DE CARRERA

DEPARTAMENTO DE INGENIERIA MECANICA

SIMULACION VIRTUAL DE UNA  
SUSPENSION DOBLE *WISHBONE* EN  
ENTORNO VRML

Enero 2010

Autor:

D. Miguel A. Tornel Fdez.

Tutoras:

Dra. Dña. Beatriz López Boada

Dra. Dña. María Jesús López Boada

# **INDICE DE CONTENIDOS**

1. INTRODUCCION .....	1
1.1. Objetivo del proyecto .....	2
1.2. Estructura del proyecto .....	2
2. SISTEMA DE SUSPENSION .....	3
2.1. Introducción.....	3
2.1.1. Movimientos de la carrocería .....	4
2.2. Elementos de la suspensión.....	5
2.2.1. Elementos elásticos de la suspensión .....	5
2.2.2. Elementos amortiguadores de la suspensión .....	8
2.2.3. Otros elementos.....	13
2.3. Tipos de suspensión.....	16
2.3.1. Suspensiones rígidas .....	17
2.3.2. Suspensiones semirrígidas .....	19
2.3.3. Suspensiones independientes.....	22
2.4. Desgaste del sistema de suspensión.....	31
3. DISEÑO ASISTIDO POR ORDENADOR EN 3D CON “SOLID EDGE” .....	33
3.1. Representación gráfica .....	33
3.1.1. Gráficos orientados a objetos .....	34
3.2. <i>Solid Edge</i> .....	35
3.3. Utilización de <i>Solid Edge</i> .....	37
3.3.1. Entorno.....	38
3.3.2. Diseño de las piezas de la suspensión.....	39

4. LA REALIDAD VIRTUAL .....	47
4.1. Historia .....	48
4.2. Características de un sistema de realidad virtual .....	49
4.2.1. Tipos de realidad virtual.....	49
4.3. Equipos utilizados para realidad virtual.....	50
4.3.1. Para visión .....	51
4.3.2. Para interactuar .....	51
4.3.3. Para audición.....	52
4.4. Ejemplos de aplicación de la realidad virtual .....	52
4.4.1. Realidad virtual en la física.....	53
4.4.2. Realidad virtual en la ingeniería.....	53
4.4.3. Realidad virtual en ciencias de la tierra .....	55
4.4.4. Realidad virtual en la oceanología .....	56
4.4.5. Realidad virtual en la medicina .....	56
4.4.6. Realidad virtual en el entretenimiento .....	58
4.4.7. Realidad virtual en el arte .....	59
4.4.8. Realidad virtual en la enseñanza.....	61
4.5. Futuro de la realidad virtual .....	62
5. EL LENGUAJE VRML .....	63
5.1. Versiones de VRML .....	64
5.2. Herramientas necesarias para trabajar con VRML .....	65
5.3. Metodología de trabajo con VRML.....	66
5.4. Programación en VRML .....	68
5.4.1. Normas básicas para crear un mundo virtual .....	68
5.4.2. Estructura básica de VRML.....	69
5.4.3. Nodos y comandos.....	71

6. CREACION DE LA SUSPENSION EN EL MUNDO VIRTUAL .....	87
6.1. Introducción de las piezas en el mundo virtual .....	90
6.2. Movimientos de la suspensión .....	96
7. CONCLUSIONES Y TRABAJOS FUTUROS .....	98
7.1. Conclusiones .....	98
7.2. Trabajos futuros.....	99
8. BIBLIOGRAFIA .....	100

# INDICE DE FIGURAS

Fig. 1 – Oscilaciones en el automóvil [2] .....	5
Fig. 2 – Ballesta.....	6
Fig. 3 – Muelles helicoidales.....	6
Fig. 4 – Fuerzas sobre un muelle .....	7
Fig. 5 – Tipos de muelles .....	7
Fig. 6 – Barra de torsión .....	8
Fig. 7 – Montaje de la barra de torsión [2].....	8
Fig. 8 – Amortiguador [1] .....	8
Fig. 9 - Amortiguador hidráulico telescópico actual .....	9
Fig. 10 - Flujo de aceite a través del paso permanente. [1] .....	10
Fig. 11 - Flujo de aceite a través de las válvulas de apertura por presión [1] .....	10
Fig. 12 – Explosión del pistón de un amortiguador monotubo.....	11
Fig. 13 - Diagrama fuerza - velocidad de un amortiguador regulable [1] .....	12
Fig. 14 – Amortiguador de gas [1] .....	12
Fig. 15 – Principio de funcionamiento de la barra estabilizadora.....	13
Fig. 16 – Conjunto Barra estabilizadora – Suspensión [2] .....	14
Fig. 17 – Mangueta y buje .....	15
Fig. 18 – Trapecios de suspensión.....	15
Fig. 19 – Suspensión rígida .....	16
Fig. 20 – Suspensión independiente .....	17
Fig. 21 – Suspensión rígida para eje trasero propulsor [2] .....	18
Fig. 22 – Suspensión trasera con muelles .....	18
Fig. 23 – Barra Panhard [2].....	19
Fig. 24 – Esquema de una suspensión De Dion [2] .....	20
Fig. 25 – Suspensión De Dion con ballestas [2] .....	20

Fig. 26 – Traviesa en forma de “U” [2] .....	21
Fig. 27 – Suspensión de eje torsional [2].....	21
Fig. 28 – Esquema de una suspensión de eje oscilante [2] .....	23
Fig. 29 – Esquema de una suspensión de eje oscilante y compensado [2].....	23
Fig. 30 – Sistema de brazo semi-arrastrado [2].....	24
Fig. 31 – Sistema de brazo arrastrado [2] .....	24
Fig. 32 – Suspensión de brazos tirados [2] .....	25
Fig. 33 – Esquema de un sistema de suspensión McPherson [2].....	25
Fig. 34 – Despiece de una suspensión McPherson [2] .....	26
Fig. 35 – Modelos de conjunto soporte-amortiguador para suspensión McPherson [2] .....	27
Fig. 36 – Suspensión de paralelogramo deformable [2] .....	28
Fig. 37 – Trapecios de la suspensión de paralelogramo deformable [2] .....	29
Fig. 38 – Esquema de una suspensión multibrazo delantera [2] .....	30
Fig. 39 – Esquema de una suspensión multibrazo trasera [2] .....	31
Fig. 40 – Ventana inicial de <i>Solid Edge</i> .....	37
Fig. 41 – Entorno de <i>Solid Edge</i> .....	38
Fig. 42 – Herramientas para la creación del boceto.....	41
Fig. 43 – Sujeción base .....	42
Fig. 44 – Creación sujeción base I .....	42
Fig. 45 – Creación sujeción base II .....	42
Fig. 46 – Creación sujeción base III .....	42
Fig. 47 – Creación sujeción base IV .....	43
Fig. 48 – Creación sujeción base V .....	43
Fig. 49 – Creación sujeción base VI .....	43
Fig. 50 – Creación sujeción base VII .....	44
Fig. 51 – Creación sujeción base VIII .....	44
Fig. 52 – Creación sujeción base IX .....	44

Fig. 53 – Creación sujeción base X .....	44
Fig. 54 – Sujeción base .....	45
Fig. 55 – Sujeción base .....	45
Fig. 56 – Mangueta.....	45
Fig. 57 – Sujeción superior .....	46
Fig. 58 – Sujeción amortiguador .....	46
Fig. 59 – Muelle .....	46
Fig. 60 – Amortiguador.....	46
Fig. 61 – Tornillo amortiguador.....	46
Fig. 62 – Tuerca amortiguador .....	46
Fig. 63 – Tornillo sujeción base .....	46
Fig. 64 – Tuerca sujeción base.....	46
Fig. 65 – Realidad virtual inmersiva .....	50
Fig. 66 – Fluido de partículas.....	53
Fig. 67 – Simulación de una erupción volcánica [14] .....	55
Fig. 68 – Modelado de relieves topográficos [14].....	56
Fig. 69 – Simulación de la superficie del océano [14] .....	56
Fig. 70 – Realidad virtual en procesos de rehabilitación.....	57
Fig. 71 – Realidad virtual en la cirugía.....	58
Fig. 72 – Realidad virtual en los videojuegos .....	58
Fig. 73 – Visita virtual de un templo.....	60
Fig. 74 – Edificio virtual creado con VRML.....	61
Fig. 75 – Primitiva <i>Box</i> [16].....	71
Fig. 76 – Primitiva <i>Sphere</i> [16].....	71
Fig. 77 – Primitiva <i>Cone</i> [16].....	72
Fig. 78 – Primitiva <i>Cylinder</i> [16].....	72
Fig. 79 – Cascada de eventos. ....	84

Fig. 80 – Sujeción base .....	88
Fig. 81 – Sujeción amortiguador .....	88
Fig. 82 – Muelle .....	88
Fig. 83 – Amortiguador.....	89
Fig. 84 – Mangueta.....	89
Fig. 85 – Sujeción superior .....	89
Fig. 86 – Pantalla inicial de la simulación .....	96
Fig. 87 – Posición de compresión y relajación durante el recorrido del muelle .....	97
Fig. 88 – Posición de compresión máxima del muelle .....	97



## **1.- INTRODUCCION**

Las Nuevas Tecnologías aportan un nuevo reto al sistema educativo, que consiste en pasar de un modelo unidireccional de formación, donde, por lo general, los saberes recaen en el profesor o en los libros de texto, a modelos más abiertos y flexibles, donde la información situada en grandes bases de datos, tiende a ser compartida entre diversos alumnos.

Frente a los modelos tradicionales de comunicación que se dan en nuestra cultura escolar, algunas tecnologías proponen nuevas alternativas en las que el aula ahora es un lugar en el que el alumno puede interactuar con otros compañeros y profesores que no tienen por qué estar situados en un mismo contexto espacial.

La Realidad Virtual es una tecnología especialmente adecuada para la enseñanza, debido a su facilidad para captar la atención de los estudiantes mediante la inmersión del alumno en mundos virtuales relacionados con cualquier rama del aprendizaje. Lo que puede ayudar en la enseñanza de los contenidos de cualquier materia.

## **1.1.- Objetivo del proyecto**

El objetivo principal del presente proyecto es realizar el estudio del movimiento de la suspensión de un vehículo, mediante técnicas virtuales.

Con ello, se pretende facilitar la comprensión del comportamiento de estos elementos de seguridad pasiva, motivando y atrayendo la atención de los estudiantes, a través de gráficos tridimensionales de calidad y el alto grado de interactividad ofrecida por los sistemas virtuales.

Para ello, se integran las herramientas de diseño asistido por ordenador en 3D, como el programa *Solid Edge*, con las que se diseñarán las piezas de la suspensión, con las simulaciones virtuales mediante el sistema VRML, el cual permite, con una simple conexión a Internet, la construcción de dichos espacios virtuales.

## **1.2.- Estructura del proyecto**

El presente proyecto consta de ocho capítulos, siendo este primero en el que se hace una introducción y se marcan los objetivos del mismo.

En el capítulo 2 se define en qué consiste un sistema de suspensión, sus partes, los diferentes tipos que hay y su funcionamiento.

El capítulo 3 explica qué es *Solid Edge* y describe cómo se han creado las piezas que componen la suspensión objeto de este proyecto.

En el capítulo 4 se hace una descripción detallada de en qué consiste la Realidad Virtual y sus diferentes usos dentro de la enseñanza.

El capítulo 5 desarrolla el funcionamiento del lenguaje VRML y explica la metodología y la programación del mismo para la creación de los mundos virtuales.

En el capítulo 6 se describe cómo se realiza el ensamblado de cada una de las piezas que componen la suspensión doble *Wishbone*, objeto de este proyecto, así como el proceso de simulación del movimiento de la misma.

El capítulo 7 indica algunas de las conclusiones a las que se ha llegado durante la realización del proyecto, y define posibles futuras simulaciones virtuales.

## **2.- SISTEMA DE SUSPENSION**

### **2.1.- Introducción**

Se llama suspensión, en un vehículo, al conjunto de elementos elásticos que se interponen entre los órganos suspendidos (bastidor, carrocería, pasajeros y carga) y los órganos no suspendidos (ruedas y ejes).

Su misión es absorber las reacciones producidas en las ruedas por las desigualdades del terreno, asegurando así la comodidad del conductor y pasajeros del vehículo y, al mismo tiempo, mantener la estabilidad y direccionalidad de este, para que mantenga la trayectoria deseada por el conductor [2].

Además, también es necesario que cumplan con otras funciones complementarias:

- Transmitir las fuerzas de aceleración y de frenada entre los ejes y bastidor.
- Resistir el par motor y de frenada.
- Resistir los efectos de las curvas.
- Conservar el ángulo de dirección en todo el recorrido.
- Conservar el paralelismo entre los ejes y la perpendicularidad del bastidor.

- Proporcionar una estabilidad adecuada al eje de balanceo.
- Aguantar la carga del vehículo.

Cuando el vehículo circula por un terreno irregular, las ruedas están sometidas a una serie de impactos que se transmiten a la carrocería a través de los elementos de unión. Si el terreno es llano, las pequeñas irregularidades del mismo son absorbidas por la elasticidad de los neumáticos. Cuando las irregularidades son grandes, los impactos producidos serían acusados por los ocupantes del vehículo, de no mediar la suspensión; la unión elástica que esta supone es capaz de absorber dichas reacciones [2].

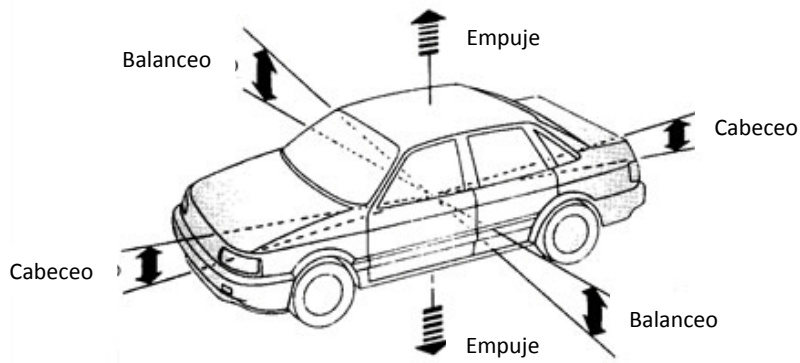
La absorción de estas reacciones se consigue por la acción combinada de los neumáticos, la elasticidad de los asientos y el sistema de suspensión.

### 2.1.1.- Movimientos de la carrocería

Cuando un automóvil pasa sobre un resalte o sobre un hoyo, se produce un golpe sobre la rueda que se transmite por medio de los ejes al chasis y que se traduce en oscilaciones.

Una mala conducción o un reparto desequilibrado de las cargas pueden también originar "oscilaciones". Estos movimientos se generan en el centro de gravedad del vehículo y se propagan en distintos sentidos. Los tres tipos de oscilaciones existentes son los que se observan en la figura 1 [2]:

- Empuje: se produce al pasar por terreno ondulado.
- Cabeceo: debido a las frenadas bruscas.
- Balanceo: se genera al tomar curvas a alta velocidad.



**Fig. 1 – Oscilaciones en el automóvil [2]**

## 2.2.- Elementos de la suspensión

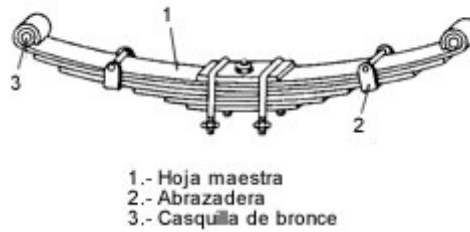
El sistema de suspensión está compuesto por un elemento flexible o elástico (muelle de ballesta o helicoidal, barra de torsión, muelle de goma, gas o aire) y un elemento de amortiguación (amortiguador).

Los elementos flexibles, como todos los muelles, tienen excelentes propiedades elásticas pero poca capacidad de absorción de energía mecánica. Es por este motivo por lo que no pueden ser montados sólo en la suspensión; necesitan el montaje de un elemento que absorba las oscilaciones producidas en su deformación. Debido a esto, los resortes se montan siempre con un amortiguador de doble efecto que absorba parte de la energía tanto en su compresión como en su expansión [2].

### 2.2.1.- Elementos elásticos de la suspensión

#### *2.2.1.1.- Ballestas*

Las ballestas (Fig. 2) están constituidas por un conjunto de hojas o láminas de acero especial para muelles, unidas mediante unas abrazaderas (2) que permiten el deslizamiento entre las hojas cuando estas se deforman por el peso que soportan. La hoja superior (1), llamada hoja maestra, va curvada en sus extremos formando unos ojos en los que se montan unos casquillos de bronce (3) para su acoplamiento al soporte del bastidor por medio de unos pernos o bulones.



**Fig. 2 – Ballesta**

El número de hojas y el espesor de las mismas son función de la carga que han de soportar. En algunos vehículos, sobre todo en camiones, además de servir de elementos de empuje, absorben con su deformación longitudinal la reacción en la propulsión [2].

#### *2.2.1.2.- Muelles helicoidales*

Los muelles helicoidales, como los que aparecen en la figura 3, se utilizan modernamente en casi todos los turismos en sustitución de las ballestas, debido a que tienen la ventaja de conseguir una elasticidad blanda debido al gran recorrido del resorte sin apenas ocupar espacio ni incrementar el peso [2].

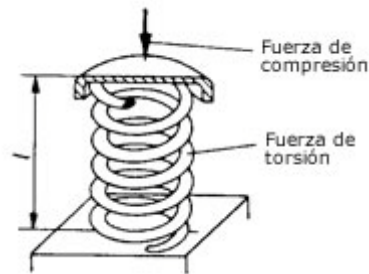


**Fig. 3 – Muelles helicoidales**

Consisten en un arrollamiento helicoidal de acero elástico formado con hilo de diámetro variable (de 10 a 15 mm). Este diámetro varía en función de la carga que tienen que soportar. Las últimas espiras son planas para facilitar el asiento del muelle sobre sus bases de apoyo.

No pueden transmitir esfuerzos laterales, y requieren, por tanto, en su montaje, bielass de empuje lateral y transversal para la absorción de las reacciones

de la rueda. Trabajan a torsión, retorciéndose proporcionalmente al esfuerzo que tienen que soportar (Fig. 4), acortando su longitud y volviendo a su posición de reposo cuando cesa el efecto que produce la deformación.



**Fig. 4 – Fuerzas sobre un muelle**

La flexibilidad de los muelles está en función del número de espiras, del diámetro del resorte, del paso entre espiras, del espesor o diámetro del hilo, y de las características del material. Se pueden conseguir muelles con una flexibilidad progresiva, utilizando diferentes diámetros de enrollado por medio de muelles helicoidales cónicos, por medio de muelles con paso entre espiras variable o disponiendo de muelles adicionales (Fig. 5) [2].



**Fig. 5 – Tipos de muelles**

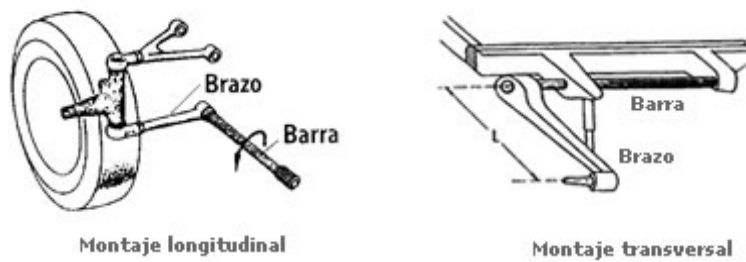
#### *2.2.1.3.- Barra de torsión.*

Este tipo de resorte está basado en el principio de que si a una varilla de acero elástico sujeta por uno de sus extremos se le aplica por el otro un esfuerzo de torsión, esta varilla tenderá a retorcerse, volviendo a su forma primitiva por su elasticidad cuando cesa el esfuerzo de torsión (Fig. 6).



**Fig. 6 – Barra de torsión**

El montaje de estas barras sobre el vehículo se realiza (Fig. 7) fijando uno de sus extremos al chasis o carrocería, de forma que no pueda girar en su soporte, y en el otro extremo, se coloca una palanca solidaria a la barra unida en su extremo libre al eje de la rueda. Cuando esta sube o baja por efecto de las desigualdades del terreno, se produce en la barra un esfuerzo de torsión cuya deformación elástica permite el movimiento de la rueda.



**Fig. 7 – Montaje de la barra de torsión [2]**

### 2.2.2.- Elementos amortiguadores de la suspensión

Los amortiguadores (Fig. 8) son los elementos encargados de absorber las vibraciones de los elementos elásticos (muelles, ballestas, barras de torsión), convirtiendo en calor la energía generada por las oscilaciones.



**Fig. 8 – Amortiguador [1]**



Cuando la rueda encuentra un obstáculo o bache, el muelle se comprime o se estira, recogiendo la energía mecánica producida por el choque. Energía que devuelve a continuación, por efecto de su elasticidad, rebotando sobre la carrocería. Este rebote en forma de vibración es el que tiene que frenar el amortiguador, recogiendo, en primer lugar, el efecto de compresión y luego el de reacción del muelle, actuando de freno en ambos sentidos; por esta razón reciben el nombre de amortiguadores de doble efecto.

Los amortiguadores pueden ser "fijos" y "regulables". Los primeros tienen siempre la misma dureza y los segundos pueden variarla dentro de unos márgenes. En los modelos más modernos este reglaje se puede hacer incluso desde el interior del vehículo [1].

#### *2.2.2.1.- Amortiguador hidráulico telescópico*

Actualmente y desde hace unos años atrás se ha impuesto en la industria el uso de los amortiguadores hidráulicos. En estos, la fuerza amortiguadora es función creciente con la velocidad.

Han existido otros tipos de amortiguadores hidráulicos, como son los giratorios y los de pistón, pero actualmente apenas se utilizan, y los que lo hacen son los de tipo telescópico, que son los que se van a presentar con más detalle y a los que se hará mención de aquí en adelante (Fig. 9).

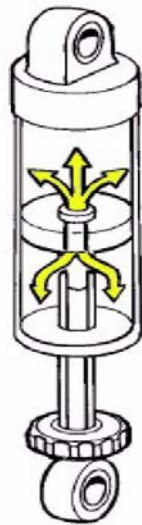


**Fig. 9 - Amortiguador hidráulico telescópico actual**

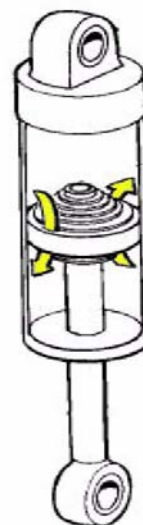
Básicamente, los amortiguadores hidráulicos telescópicos constan de un pistón que trabaja dentro de un cilindro en el que hay aceite. Sobre el pistón existen una serie de orificios y unas válvulas precomprimidas que permiten el paso de aceite de una parte a otra del pistón cuando la presión supera un valor

dado. Los orificios representan el paso permanente y las válvulas el paso de apertura por presión respectivamente.

El paso permanente (Fig. 10) viene a ser orificios fijos (salvo cuando el amortiguador es un mono-tubo regulable, en cuyo caso la regulación suele consistir en la variación del tamaño del orificio) que restringen el flujo del caudal. En el caso de las válvulas de apertura por presión (Fig. 11), para que estas se abran es necesario ejercer sobre ellas una determinada presión y a medida que esta aumenta la apertura va siendo mayor [1].

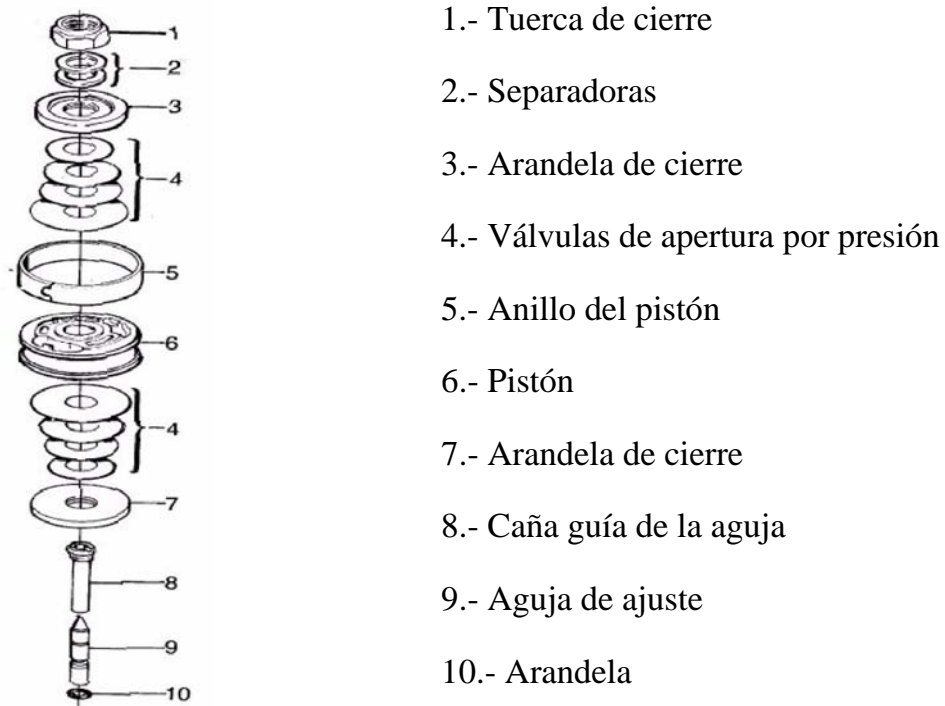


**Fig. 10 - Flujo de aceite a través del paso permanente. [1]**



**Fig. 11 - Flujo de aceite a través de las válvulas de apertura por presión [1]**

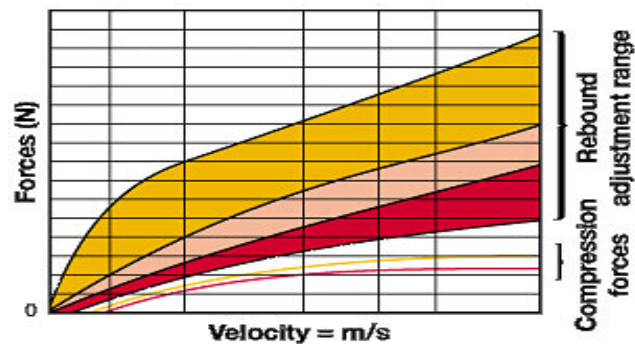
En la figura 12 se puede ver en detalle el pistón con los discos que componen las válvulas de apertura y la aguja que regula el diámetro de la válvula de apertura por área cuando el amortiguador es variable. Si se mueve la rosca de ajuste de la dureza del amortiguador para “ablandarlo”, la aguja abre el orificio facilitando el paso del aceite y disminuyendo la rigidez del amortiguador y viceversa.



**Fig. 12 – Explosión del pistón de un amortiguador monotubo**

Cuando la velocidad entre ambos extremos del amortiguador es baja, las válvulas de apertura por presión permanecen cerradas y el aceite pasa a través de los orificios del paso permanente. Una vez la presión del aceite alcanza la presión de tarado de las válvulas de presión, estas empiezan a abrirse y dejan pasar el aceite. Cuando más aumenta la presión, las válvulas se abren más hasta que su apertura es completa y la ley de fuerza en el amortiguador queda controlada nuevamente por el paso del aceite a través del orificio de paso permanente.

Lo normal es que las válvulas de extensión y compresión sean diferentes, lo que posibilita que el esfuerzo en compresión sea menor para una misma velocidad. En los vehículos de carretera, interesa utilizar una característica de amortiguamiento más blanda en compresión (Fig. 13). Esto se hace para evitar la transmisión a través del amortiguador de las grandes fuerzas compresivas que se generarían en el mismo cuando la rueda se encuentra con un obstáculo [1].

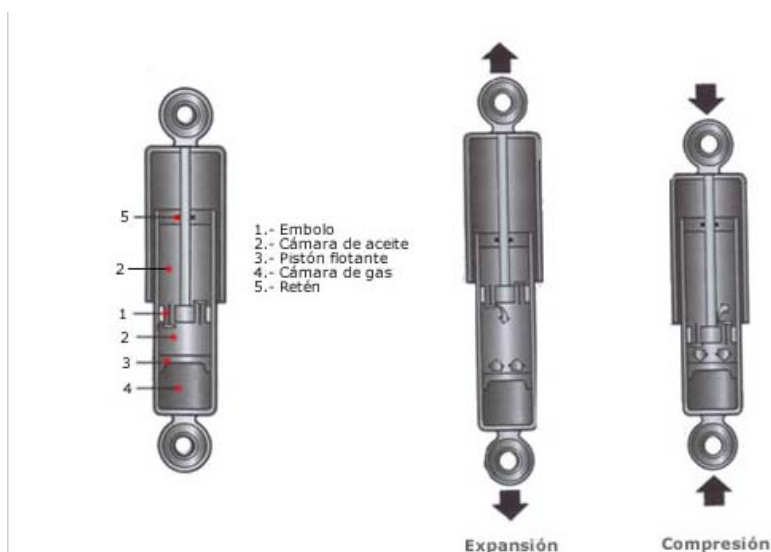


**Fig. 13 - Diagrama fuerza - velocidad de un amortiguador regulable [1]**

#### 2.2.2.2.- Amortiguador de gas

Los amortiguadores de gas, como los de la figura 14, trabajan bajo el mismo principio básico que los hidráulicos, pero contienen en uno de sus extremos nitrógeno a alta presión (aproximadamente 25 bares).

Un pistón flotante separa este gas del aceite impidiendo que se mezclen. Cuando el aceite, al desplazarse el vástago, comprime el gas, este sufre una variación de volumen que permite dar una respuesta instantánea y un funcionamiento silencioso. Los amortiguadores de gas además de amortiguar también hacen en cierto modo de resorte elástico, es por ello que este tipo de amortiguadores vuelven a su posición cuando se deja de actuar sobre ellos [1].



**Fig. 14 – Amortiguador de gas [1]**

- Amortiguadores de gas no regulables: suelen ser amortiguadores monotubo o bitubo, muy resistentes a golpes, de alta duración y de alta resistencia a la pérdida de eficacia por temperatura de trabajo. Aunque el precio es mayor, se ve compensado por su durabilidad y fiabilidad. Es un tipo de amortiguador de muy alta calidad. Su uso es ciertamente recomendable para los vehículos de altas prestaciones.
- Amortiguadores de gas regulables: Son amortiguadores monotubo, con o sin botella exterior, con posibilidad de variación de tarados. Es un tipo de amortiguador de alta tecnología, con precio alto pero proporcional a su eficacia, por eso es el más usado en conducción deportiva, en los vehículos de competición y de altas prestaciones.

### 2.2.3.- Otros elementos

#### *2.2.3.1.- Barras estabilizadoras*

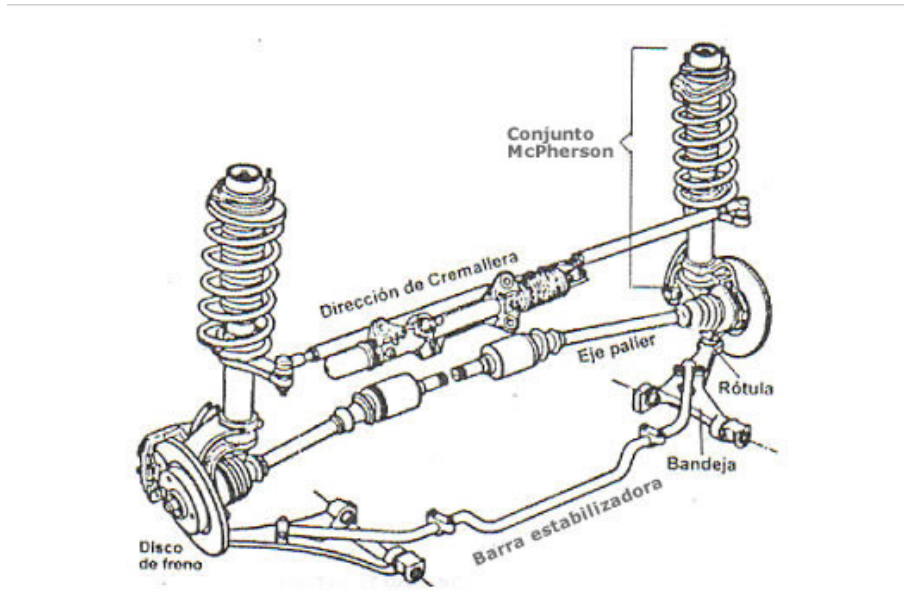
Las barras estabilizadoras funcionan como se muestra en la figura 15. Cuando un vehículo toma una curva, por la acción de la fuerza centrífuga, se carga el peso del coche sobre las ruedas exteriores, con lo cual la carrocería tiende a inclinarse hacia ese lado con peligro de vuelco y la correspondiente molestia para sus ocupantes.

Para evitar estos inconvenientes, las barras estabilizadoras (Fig. 16) se montan sobre los ejes delantero y trasero, y consisten esencialmente en una barra de acero cuyos extremos se fijan a los soportes de suspensión de las ruedas. De esta forma, al tomar una curva, como una de las ruedas tiende a bajar y la otra a subir, se crea un par de torsión en la barra que absorbe el esfuerzo y se opone a que esto ocurra. Así, impide que la carrocería se incline a un lado, manteniéndola estable [2].



**Fig. 15 – Principio de funcionamiento de la barra estabilizadora**

El mismo efecto se produce cuando una de las ruedas encuentra un bache u obstáculo, creando, al bajar o subir la rueda, un par de torsión en la barra que hace que la carrocería se mantenga en posición horizontal. En caso de circular en línea recta y en condiciones normales la acción de la barra es nula.



**Fig. 16 – Conjunto Barra estabilizadora – Suspensión [2]**

#### 2.2.3.2.- *Silentblocks y cojinetes elásticos*

Los *silentblocks* y cojinetes elásticos son aislantes de caucho u otro material elastómero que se encargan de amortiguar las reacciones en los apoyos de la suspensión. Su misión es amortiguar los golpes existentes entre dos elementos en los que existe movimiento.

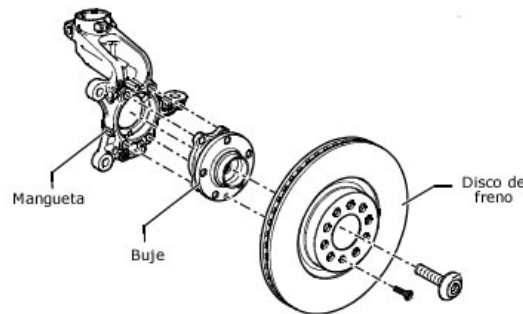
Los cojinetes elásticos son elementos de caucho que permiten la unión de los componentes de la suspensión facilitando un pequeño desplazamiento.

#### 2.2.3.3.- *Rótulas*

Las rótulas constituyen un elemento de unión y fijación de la suspensión y de la dirección, que permite su pivotamiento y giro manteniendo la geometría de las ruedas.

#### 2.2.3.4.- Mangueta y buje

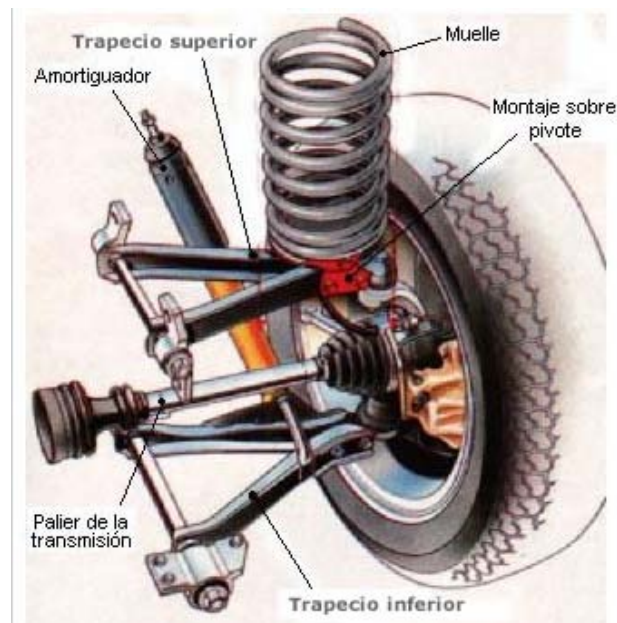
La mangueta de la suspensión (Fig. 17) es una pieza fabricada con acero o aleaciones que une el buje de la rueda (Fig. 17) y la rueda a los elementos de la suspensión: tirantes, trapecios, amortiguador, etc.



**Fig. 17 – Mangueta y buje**

#### 2.2.3.5.- Trapecios o brazos de suspensión.

Los trapecios (Fig. 18) son brazos articulados, fabricados en fundición o en chapa de acero embutida, que soportan al vehículo a través de la suspensión. Unen la mangueta y su buje al vehículo mediante elementos elásticos (*silentblocks*) y elementos de guiado (rótulas), soportando los esfuerzos generados por este en su funcionamiento.



**Fig. 18 – Trapecios de suspensión**



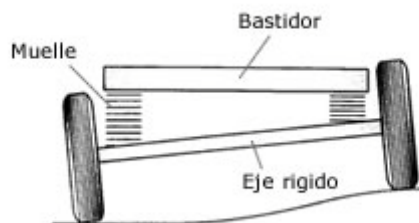
### 2.3.- Tipos de suspensión

Según el tipo de elementos empleados y la forma de montaje de los mismos, existen varios sistemas de suspensión, todos ellos basados en el mismo principio de funcionamiento. Constan de un sistema elástico, amortiguación y barra estabilizadora independientes para cada uno de los ejes del vehículo.

Actualmente existen distintas disposiciones de suspensión cuyo uso depende del tipo de comportamiento que se busca en el vehículo: mayores prestaciones, más comodidad, sencillez y economía, etc.

Las primeras suspensiones estaban formadas por un "eje rígido" en cuyos extremos se montaban las ruedas. Como consecuencia de ello, todo el movimiento que afecta a una rueda se transmite a la otra del mismo eje. En la figura 19 se ve cómo al elevarse una rueda, se extiende su inclinación al eje y de este a la otra rueda. Como el eje va fijado directamente sobre el bastidor, la inclinación se transmite a todo el vehículo.

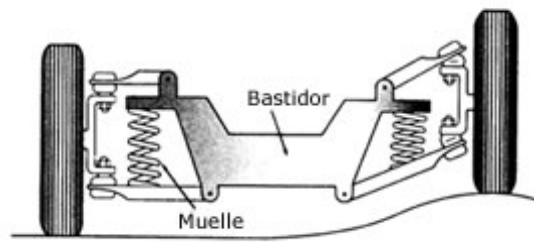
Este montaje es muy resistente y más económico de fabricar, pero tiene la desventaja de ser poco cómodo para los pasajeros y tener una menor seguridad.



**Fig. 19 – Suspensión rígida**

El sistema de suspensión "independiente" (Fig. 20) tiene un montaje elástico independiente que no está unido a otras ruedas. A diferencia del sistema rígido, el movimiento de una rueda no se transmite a la otra y la carrocería resulta menos afectada.



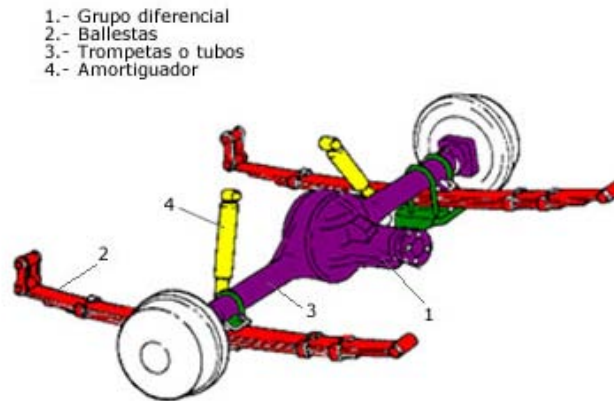


**Fig. 20 – Suspensión independiente**

### 2.3.1.- Suspensiones rígidas

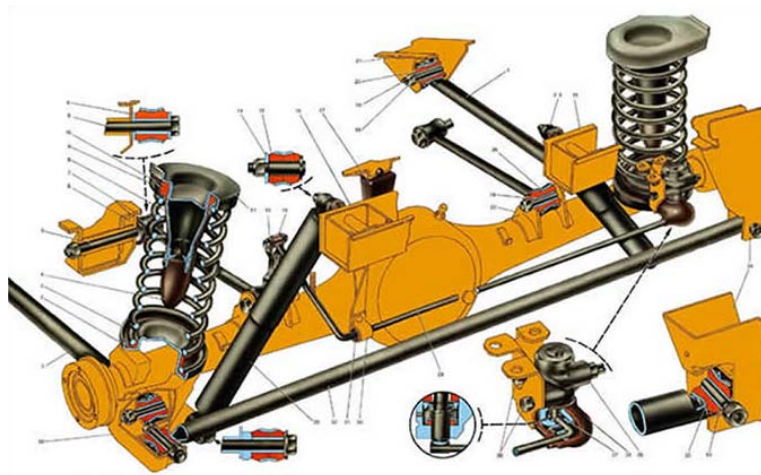
La suspensión rígida tiene unidas las ruedas mediante un eje rígido formando un conjunto. Presenta el inconveniente de que al estar unidas ambas ruedas, las vibraciones producidas por la acción de las irregularidades del pavimento, se transmiten de un lado al otro del eje. Además, el peso de las masas no suspendidas aumenta notablemente debido al peso del eje rígido y al peso del grupo cónico diferencial en los vehículos de tracción trasera. En estos últimos, el grupo cónico sube y baja en las oscilaciones como una parte integradora del eje rígido. Como principal ventaja, los ejes rígidos destacan por su sencillez de diseño y no producen variaciones significativas en los parámetros de la rueda como caída, avance, etc. El principal uso de esta disposición de suspensión se realiza sobre todo en vehículos industriales, autobuses, camiones y vehículos todo terreno [2].

En la figura 21 se muestra un modelo de eje rígido actuando de eje propulsor. En estos casos el eje está constituido por una caja que contiene el mecanismo diferencial (1) y por los tubos (3) que contienen los palieres. El eje rígido en este caso se apoya contra el bastidor mediante ballestas (2) que hacen de elemento elástico transmitiendo las oscilaciones. Completan el conjunto los amortiguadores (4).



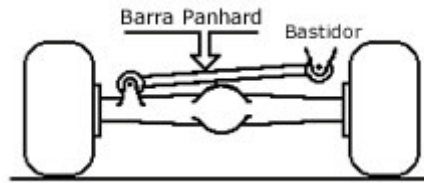
**Fig. 21 – Suspensión rígida para eje trasero propulsor [2]**

En la figura 22 se muestra una suspensión rígida trasera, que sustituye las ballestas por muelles. Esta suspensión no presenta rigidez longitudinal, de forma que el eje rígido lleva incorporada barras longitudinales que mantienen el eje fijo en su posición, evitando que se mueva en el eje longitudinal.



**Fig. 22 – Suspensión trasera con muelles**

Además, para estabilizar el eje y generar un único centro de balanceo de la suspensión, se añade una barra transversal que une el eje con el bastidor. A esta barra se le conoce con el nombre de barra "Panhard" (Fig. 23). Tanto las barras longitudinales como la barra Panhard dispone de articulaciones elásticas que las unen con el eje y la carrocería [2].

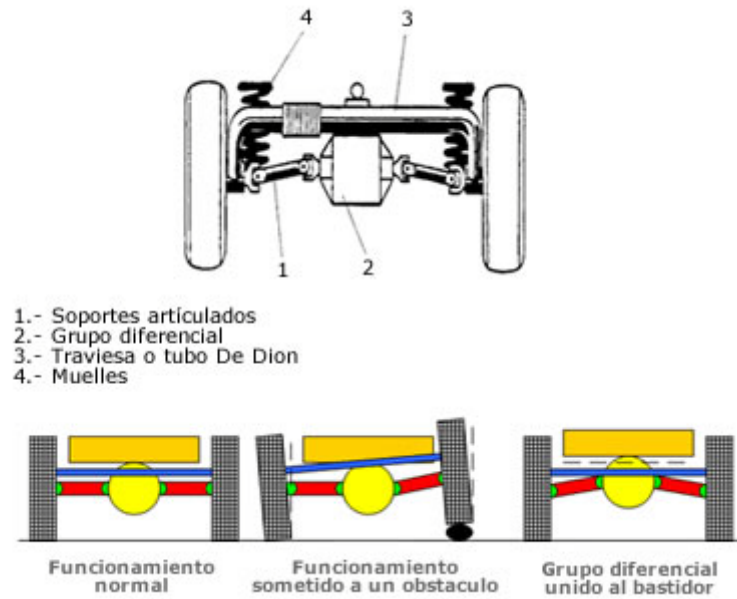


**Fig. 23 – Barra Panhard [2]**

### 2.3.2.- Suspensiones semirrígidas

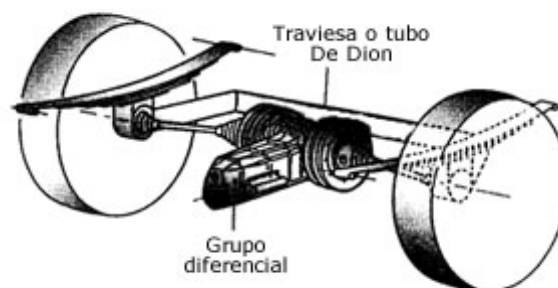
Las suspensiones semirrígidas son muy parecidas a las anteriores. Las ruedas están unidas entre sí como en el eje rígido, pero su diferencia principal es que transmiten de una forma parcial las oscilaciones que reciben de las irregularidades del terreno. En cualquier caso, aunque la suspensión no es rígida total, tampoco es independiente. La función motriz se separa de la función de suspensión y de guiado [2].

En la figura 24 se muestra una suspensión de este tipo. Se trata de una suspensión con **eje "De Dion"**. En ella, las ruedas van unidas mediante soportes articulados (1) al grupo diferencial (2), que en la suspensión con eje De Dion es parte de la masa suspendida, es decir, va anclado al bastidor del automóvil. Bajo este aspecto se transmite el giro a las ruedas a través de dos semiejes (palieres) como en las suspensiones independientes. A su vez, ambas ruedas están unidas entre sí mediante una traviesa o tubo De Dion (3) que las sujeta de forma rígida permitiendo a la suspensión deslizamientos longitudinales. Este sistema tiene la ventaja, frente al eje rígido, de que se disminuye la masa no suspendida, debido al poco peso de la traviesa del eje De Dion y al anclaje del grupo diferencial al bastidor, y mantiene los parámetros de la rueda prácticamente constantes, como los ejes rígidos, gracias al anclaje rígido de la traviesa. La suspensión posee además elementos elásticos de tipo muelle helicoidal (4) y suele ir acompañada de brazos longitudinales que limitan los desplazamientos longitudinales.



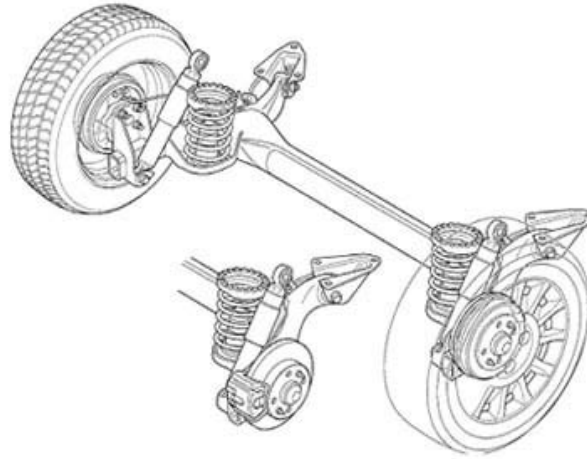
**Fig. 24 – Esquema de una suspensión De Dion [2]**

En la figura 25 se muestra otra suspensión semirrígida "De Dion", pero que utiliza ballestas en vez de muelles.



**Fig. 25 – Suspensión De Dion con ballestas [2]**

El "**eje torsional**" es otro tipo de suspensión semirrígida (semi-independiente), utilizada en las suspensiones traseras, en vehículos que tienen tracción delantera. La travesa o tubo que une las dos ruedas tiene forma de "U", como se puede observar en la figura 26, por lo que es capaz de deformarse un cierto ángulo cuando una de las ruedas encuentra un obstáculo, para después, una vez pasado el obstáculo, volver a la posición inicial.



**Fig. 26 – Traviesa en forma de “U” [2]**

Las ruedas están unidas rígidamente a dos brazos longitudinales unidos por un travesaño que los une y que se tuerce durante las sacudidas no simétricas, dando estabilidad al vehículo. Esta configuración (Fig. 27) da lugar, a causa de la torsión del puente, a una recuperación parcial del ángulo de caída de alto efecto de estabilización, características que junto al bajo peso, al bajo coste y al poco espacio que ocupan, resulta ideal para instalarla junto con otros componentes debajo del piso (depósito de combustible, escape, etc.). Esta configuración ha convertido a este tipo de suspensiones en una de las más empleadas en vehículos de gama media-baja.



**Fig. 27 – Suspensión de eje torsional [2]**

### 2.3.3.- Suspensiones independientes

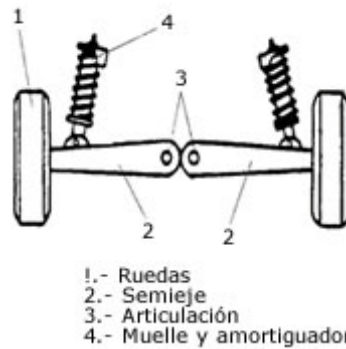
Actualmente, la suspensión independiente a las cuatro ruedas se va utilizando cada vez más debido a que es la más óptima desde el punto de vista de confort y estabilidad, al reducir de forma independiente las oscilaciones generadas por el pavimento sin transmitirlas de una rueda a otra del mismo eje. La principal ventaja añadida de la suspensión independiente es que posee menor peso no suspendido que otros tipos de suspensión por lo que las acciones transmitidas al chasis son de menor magnitud. El diseño de este tipo de suspensión debe garantizar que las variaciones de caída de rueda y ancho de ruedas en las ruedas directrices sean pequeñas para conseguir una dirección segura del vehículo. Por el contrario, para cargas elevadas esta suspensión puede presentar problemas. Actualmente este tipo de suspensión es el único que se utiliza para las ruedas directrices [2].

El número de modelos de suspensión independiente es muy amplio y además posee numerosas variantes. Los principales tipos de suspensión de tipo independiente son [2]:

- Suspensión de eje oscilante.
- Suspensión de brazos tirados.
- Suspensión McPherson.
- Suspensión de paralelogramo deformable.
- Suspensión multibrazo (*multilink*).

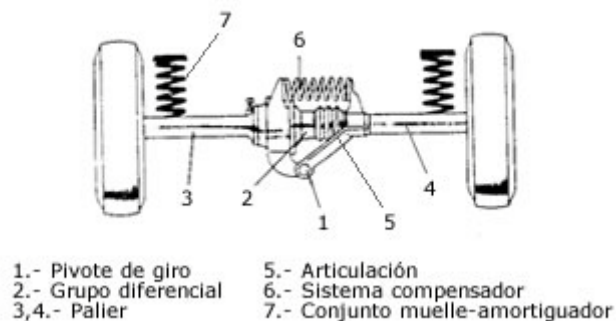
#### *2.3.3.1.- Suspensión de eje oscilante*

La peculiaridad de la suspensión de eje oscilante, que se muestra en la figura 28, es que el elemento de rodadura (1) y el semieje (2) son solidarios (salvo el giro de la rueda), de forma que el conjunto oscila alrededor de una articulación (3) próxima al plano medio longitudinal del vehículo. Este tipo de suspensión no se puede usar como eje directriz puesto que en el movimiento oscilatorio de los semiejes se altera notablemente la caída de las ruedas en las curvas. Completan el sistema de suspensión dos conjuntos muelle-amortiguador telescópico (4).



**Fig. 28 – Esquema de una suspensión de eje oscilante [2]**

Una variante de este sistema es el realizado mediante un eje oscilante pero de una sola articulación, mostrado en la figura 29. La ventaja que presenta es que el pivote de giro (1) está a menor altura que en el eje oscilante de dos articulaciones. El mecanismo diferencial (2) oscila con uno de los palieres (3) mientras que el otro (4) se mueve a través de una articulación (6) que permite a su vez un desplazamiento de tipo axial en el árbol de transmisión. El sistema también cuenta con dos conjuntos muelle-amortiguador (7).



**Fig. 29 – Esquema de una suspensión de eje oscilante y compensado [2]**

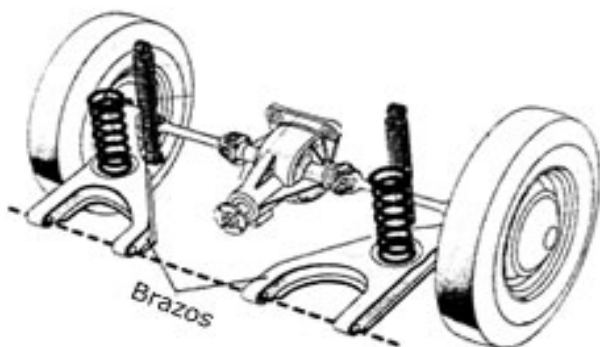
#### 2.3.3.2.- *Suspensión de brazos tirados o arrastrados*

La suspensión de brazos tirados o arrastrados se caracteriza por tener dos elementos soporte o "brazos" en disposición longitudinal que van unidos por un extremo al bastidor y por el otro a la mangueta de la rueda. Si el eje es de tracción, el grupo diferencial va anclado al bastidor. En cualquier caso, las ruedas son tiradas o arrastradas por los brazos longitudinales que pivotan en el anclaje de la carrocería.

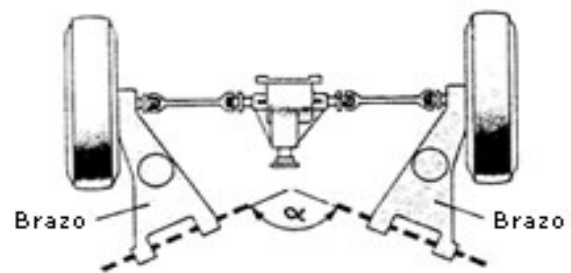


Este sistema de suspensión ha dado un gran número de variantes cuyas diferencias estriban fundamentalmente en cuál es el eje de giro del brazo tirado en el anclaje al bastidor y cuál es el elemento elástico que utiliza.

Las figuras 30 y 31 muestran cómo los brazos tirados pueden pivotar de distintas formas: en la figura 30 los brazos longitudinales pivotan sobre un eje de giro perpendicular al plano longitudinal del vehículo. Este tipo de suspensión apenas produce variaciones de vía, caída o avance de la rueda. En la figura 31 pivotan los brazos sobre ejes que tienen componentes longitudinales, es decir, sobre ejes oblicuos al plano longitudinal del vehículo. A esta última variante también se la conoce como "brazos semi-arrastrados" y tiene la ventaja de que no precisa estabilizadores longitudinales debido a la componente longitudinal que tiene el propio brazo o soporte. Aquí las variaciones de caída y de vía dependen de la posición e inclinación de los brazos longitudinales, por lo tanto, permite que se varíe durante la marcha la caída y el avance de las ruedas con lo que se mejora la estabilidad del vehículo. En cuanto al tipo de elementos elásticos que se utilizan en estas suspensiones, se encuentran las barras de torsión y los muelles.



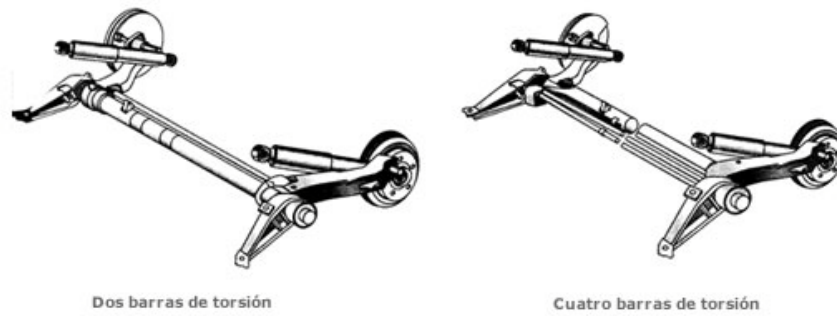
**Fig. 30 – Sistema de brazo arrastrado [2]**



**Fig. 31 – Sistema de brazo semi - arrastrado [2]**

En los sistemas de suspensión de brazos tirados con barras de torsión, como los de la figura 32, las barras se montan de manera transversal a la carrocería. Como mínimo se utilizan dos, pudiendo llegar incluso a montar cuatro en vehículos cuyo tarado deba ser mayor. Por ejemplo, existen modelos que montan dos barras de torsión en el puente trasero, mientras que un modelo similar pero con mayor motorización, monta cuatro barras.



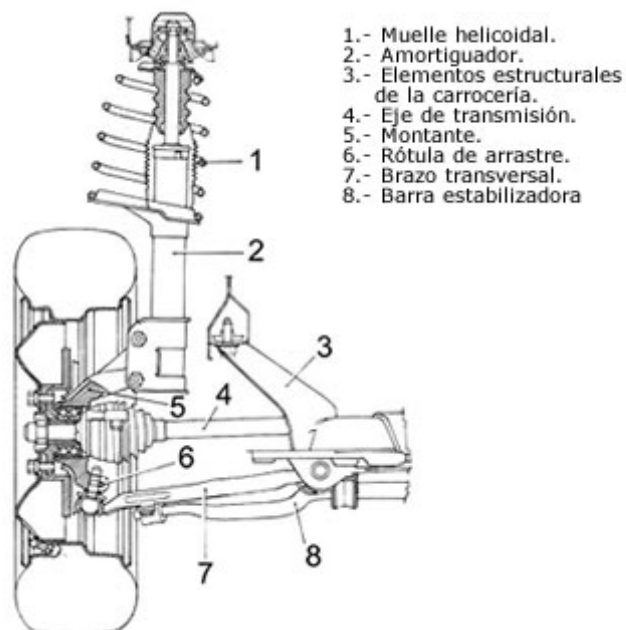


**Fig. 32 – Suspensión de brazos tirados [2]**

#### 2.3.3.3.- Suspensión McPherson

La suspensión McPherson fue desarrollada por Earle S. McPherson, ingeniero de Ford del cual recibe su nombre (Fig. 33). Este sistema es uno de los más utilizados en el tren delantero, aunque se puede montar igualmente en el trasero. Este sistema ha tenido mucho éxito, sobre todo en vehículos más modestos, por su sencillez de fabricación y mantenimiento, el coste de producción y el poco espacio que ocupa.

Con esta suspensión es imprescindible que la carrocería sea más resistente en los puntos donde se fijan los amortiguadores y muelles, con objeto de absorber los esfuerzos transmitidos por la suspensión.

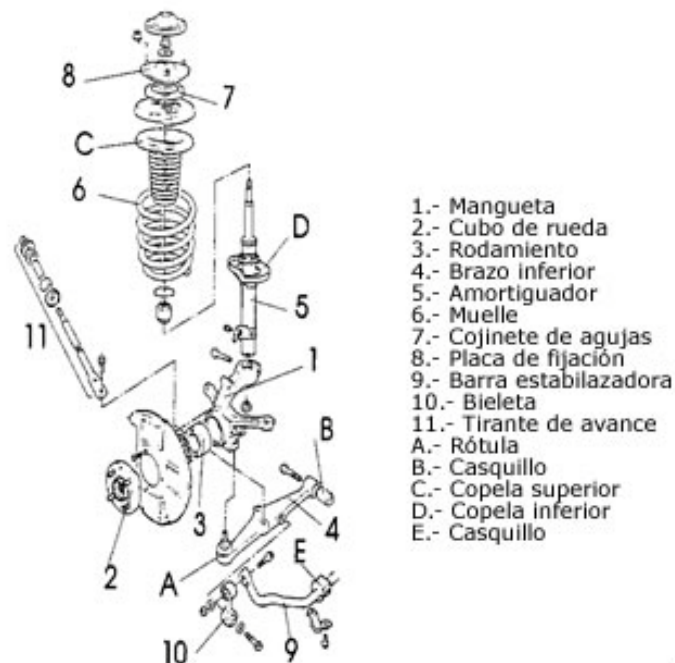


**Fig. 33 – Esquema de un sistema de suspensión McPherson [2]**

La figura 34 muestra un modelo detallado de una suspensión McPherson con brazo inferior y barra estabilizadora.

La mangueta (1) de la rueda va unida al cubo (2) permitiendo el giro de este mediante un rodamiento (3). A su vez la mangueta va unida al bastidor a través de dos elementos característicos de toda suspensión McPherson:

- El brazo inferior (4) que va unido a la mangueta (1) mediante una unión elástica (A) (rótula) y unido al bastidor mediante un casquillo (B).
- El conjunto muelle helicoidal-amortiguador: El amortiguador (5) va anclado de forma fija a la parte superior de la mangueta (1) y el muelle (6) es concéntrico al amortiguador y está sujeto mediante dos copelas superior (C) e inferior (D). El amortiguador está unido al bastidor por su parte superior mediante un cojinete de agujas (7) y una placa de fijación (8). En las ruedas delanteras se hace necesaria la existencia de este cojinete axial ya que el amortiguador al ser solidario a la mangueta gira con esta al actuar la dirección.



**Fig. 34 – Despiece de una suspensión McPherson [2]**

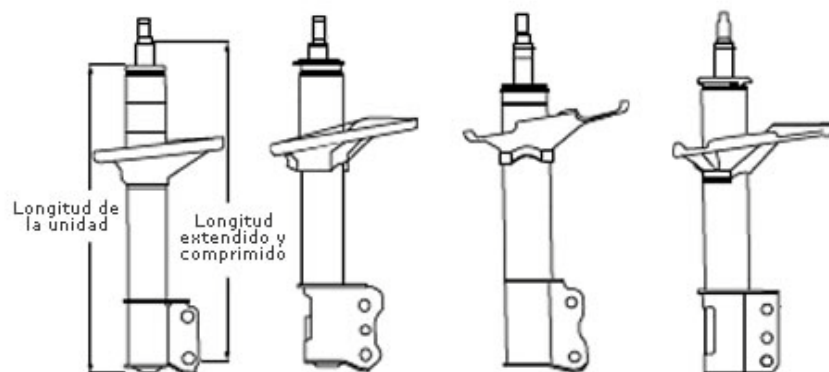
La suspensión tipo McPherson forma un mecanismo de tipo triángulo articulado formado por el brazo inferior (4), el conjunto muelle-amortiguador y el propio chasis. El lado del triángulo que corresponde al muelle-amortiguador es de compresión libre, por lo que sólo tiene un único grado de libertad: la tracción o compresión de los elementos elásticos y amortiguador. Al transmitirse a través

del muelle-amortiguador todos los esfuerzos al chasis es necesario un dimensionado más rígido de la carrocería en la zona de apoyo de la placa de fijación (8).

Como elementos complementarios a esta suspensión se encuentra la barra estabilizadora (9) unida al brazo inferior (4) mediante una bieleta (10) y al bastidor mediante un casquillo (E), y en este caso un tirante de avance (11).

Como se observa en la figura 35, actualmente existen múltiples variantes en cuanto a la sustitución del tirante inferior (4), las cuales se pueden cambiar por un triángulo inferior, doble bieleta transversal con tirante longitudinal, etc. A estos últimos sistemas también se les ha denominado "falsa" McPherson, aunque en cualquier caso todos ellos utilizan el amortiguador como elemento de guía y mantienen la estructura de triángulo articulado.

La suspensión clásica McPherson dispone de la barra estabilizadora como tirante longitudinal, mientras que las denominadas "falsa" McPherson ya absorben los esfuerzos longitudinales con la propia disposición del anclaje del elemento que sustituye al brazo inferior.

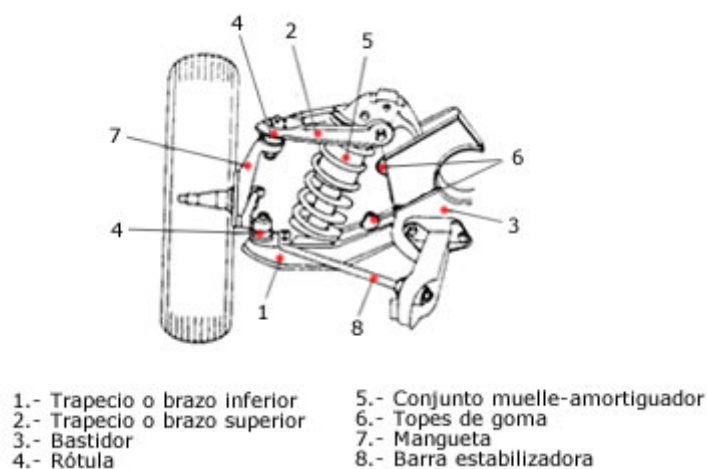


**Fig. 35 – Modelos de conjunto soporte-amortiguador para suspensión McPherson [2]**

#### 2.3.3.4.- Suspensión de paralelogramo deformable

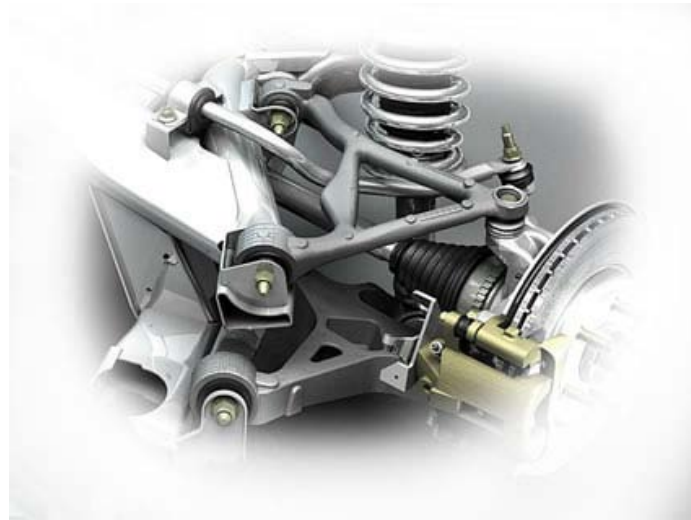
La suspensión de paralelogramo deformable junto con la McPherson es la más utilizada en un gran número de automóviles, tanto para el tren delantero como para el trasero. Esta suspensión también se denomina suspensión por trapecio articulado y suspensión de triángulos superpuestos.

En la figura 36 se muestra una suspensión convencional de paralelogramo deformable. El paralelogramo está formado por un brazo superior (2) y otro inferior (1) que están unidos al chasis a través de unos pivotes, cerrando el paralelogramo a un lado el propio chasis y al otro la propia mangueta (7) de la rueda. La mangueta está articulada con los brazos mediante rótulas esféricas (4) que permiten la orientación de la rueda. Los elementos elásticos y amortiguador coaxiales (5) son de tipo resorte helicoidal e hidráulico telescópico respectivamente y están unidos por su parte inferior al brazo inferior y por su parte superior al bastidor. Completan el sistema unos topes (6) que evitan que el brazo inferior suba lo suficiente como para sobrepasar el límite elástico del muelle y un estabilizador lateral (8) que va anclado al brazo inferior (1).



**Fig. 36 – Suspensión de paralelogramo deformable [2]**

Con distintas longitudes de los brazos (1) y (2) se pueden conseguir distintas geometrías de suspensión de forma que puede variar la estabilidad y la dirección según sea el diseño de estos tipos de suspensión (Fig. 37).



**Fig. 37 – Trapecios de la suspensión de paralelogramo deformable [2]**

La evolución de estos sistemas de suspensión de paralelogramo deformable ha llegado hasta las actuales suspensiones llamadas multibrazo o *multilink*.

#### *2.3.3.5.- Suspensiones Multibrazo o Multilink*

Las suspensiones multibrazo se basan en el mismo concepto básico que sus precursoras: las suspensiones de paralelogramo deformable, es decir, el paralelogramo está formado por dos brazos transversales, la mangueta de la rueda y el propio bastidor. La diferencia fundamental que aportan estas nuevas suspensiones es que los elementos guía de la suspensión multibrazo pueden tener anclajes elásticos mediante manguitos de goma. Gracias a esta variante las multibrazo permiten modificar tanto los parámetros fundamentales de la rueda, como la caída o la convergencia, de la forma más apropiada de cara a la estabilidad en las distintas situaciones de uso del automóvil. Esto significa que las dinámicas longitudinal y transversal pueden configurarse de forma precisa y prácticamente independiente entre sí, y que puede alcanzarse un grado máximo de estabilidad direccional y confort.

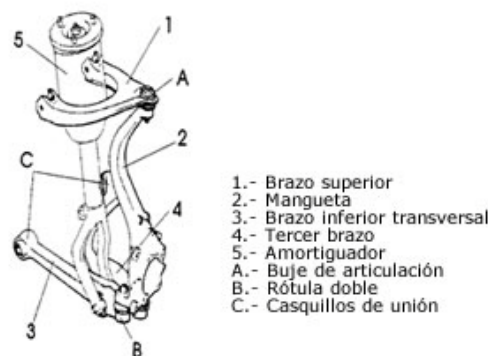
A principios de los noventa se comenzó a instalar estos sistemas multibrazo en automóviles de serie dando buenos resultados, aunque había reticencias para los ejes no motores. En la actualidad, las grandes berlinas adoptan este sistema en uno de los trenes, o en ambos. Para que una suspensión se considere multibrazo debe estar formada al menos por tres brazos.

Las suspensiones multibrazo se pueden clasificar en dos grupos fundamentales:

- Suspensiones multibrazo con elementos de guía transversales u oblicuos con funcionamiento similar al de las suspensiones de paralelogramo deformable.
- Suspensiones multibrazo que además disponen de brazos de guía longitudinal con un funcionamiento que recuerda a los sistemas de suspensión de ruedas tiradas por brazos longitudinales.

Las figuras 38 y 39 muestran un sistema multibrazo delantero y uno trasero del tipo paralelogramo deformable con tres brazos. La suspensión delantera (Figura 38) consta de un brazo superior (1) que va unido a una mangueta (2) larga y curvada mediante un buje de articulación (A) y un brazo inferior transversal (3) que va unido a la mangueta por una rótula doble (B) y al bastidor por un casquillo (C) que aísla de las vibraciones. Cierra el paralelogramo deformable el propio bastidor como en cualquier suspensión de este tipo.

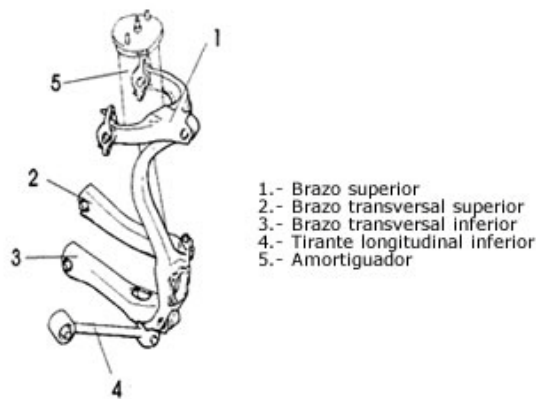
Esta suspensión dispone además de un tercer brazo (4) que hace de tirante longitudinal y que está unido al bastidor y a la mangueta de la misma forma que el brazo inferior transversal (3). La gran altura de la prolongación de la mangueta consigue una disminución de los cambios de convergencia de la rueda y un ángulo de avance negativo.



**Fig. 38 – Esquema de una suspensión multibrazo delantera [2]**

La suspensión trasera (figura 39) consta de un brazo superior (1) con forma de triángulo como la delantera, pero dispone de dos brazos transversales, superior (2) e inferior (3) y un tirante longitudinal inferior (4). Las articulaciones son similares al modelo de suspensión delantera.

Ambos sistemas poseen como elementos elásticos muelles helicoidales y amortiguadores telescópicos (5) y también barra estabilizadora. Observar que, en la disposición delantera, el amortiguador va anclado a la barra inferior transversal (3) mediante una horquilla.



**Fig. 39 – Esquema de una suspensión multibrazo trasera [2]**

## 2.4.- Desgaste del sistema de suspensión

Un sistema de suspensión mal cuidado puede hacer que un automóvil salga de la vía por muy nuevos que estén los neumáticos que lleve. Pese a la importancia del buen estado de este sistema, dice el RACE que un 10% de las ITV's desfavorables lo son por el mal estado de la suspensión. Al desconocimiento generalizado que tiene el conductor medio sobre el estado del sistema de suspensión de su vehículo, se añade la opacidad de la Administración sobre cuál debe ser el límite legal de desgaste de este sistema [6].

Normalmente el desgaste del sistema no viene dado por la rotura de los muelles, que ocurre en contadas ocasiones, sino por el envejecimiento progresivo de los amortiguadores. Cuando los amortiguadores están gastados, el fluido pasa rápidamente de una cámara a otra, el muelle bota y rebota y el vehículo se convierte en una enorme masa difícil de mantener en contacto con la vía.

Como el desgaste es progresivo, el conductor se va acostumbrando a la nueva forma de trabajar de su sistema de suspensión, de manera que aparentemente no nota nada.

El grado de desgaste de la suspensión viene dado por el uso que se le dé al sistema. A diferencia de lo que ocurre con otros sistemas del vehículo, en el caso de la suspensión no existe un manejo directo del conductor. No hay un accionamiento, como puede pasar con los frenos o la dirección, pero sí hay

formas de usarlo que llevan al envejecimiento prematuro de los elementos que componen el sistema. Pasar por un resalto a gran velocidad o una conducción agresiva por una carretera llena de baches, equivale a machacar los muelles y amortiguadores, con el consiguiente riesgo para la integridad del vehículo y sus ocupantes.

No existe para el sistema de suspensión otro mantenimiento que la revisión de los elementos que lo componen para, cuando llega el momento, proceder a su sustitución. Los sistemas de suspensión actuales se fabrican con una resistencia que no se puede comprobar “a mano”, sino que deben ser inspeccionados en un taller mecánico cada 20.000 kilómetros, ya que la vida útil de sus componentes ronda los 60.000 a 80.000 kilómetros [6].

Como síntomas de que algo no va bien en el sistema de suspensión se deben considerar [3]:

- Los neumáticos se desgastan de forma prematura e irregular, y disminuye su adherencia. Aumenta el riesgo de *aquaplaning*.
- Una pérdida de aceite en los amortiguadores.
- La distancia de frenado aumenta y la frenada se vuelve más inestable, produciéndose una excesiva inclinación del morro del coche al frenar.
- Una vibración exagerada de las luces al circular de noche, lo que puede provocar el deslumbramiento a los conductores que se acercan en sentido contrario.
- El nivel de confort de los ocupantes del vehículo se ve mermado por las sacudidas durante la conducción.
- Se acentúa la inestabilidad de la dirección y la dificultad de controlar el vehículo en las curvas.



## **CAPITULO 3. DISEÑO ASISTIDO POR ORDENADOR EN 3D CON “SOLID EDGE”**

### **3.1.- Representación gráfica**

La palabra “dibujo” deriva de un vocablo latino, “designare”, que significa designar, señalar. El dibujo consigue mostrar en un soporte físico una forma que suele corresponderse con un concepto u objeto real, aunque no siempre sea así.

El dibujo sirve como herramienta de trabajo, como boceto, apunte o recordatorio. Puede llevarse a cabo en múltiples soportes y con diversos materiales. El dibujo técnico, que consiste en un dibujo normalizado, es el procedimiento utilizado para representar topografía, trabajos de ingeniería, edificios y piezas de maquinaria. La utilización del dibujo técnico es importante en todas las ramas de la ingeniería y en la industria, y también en arquitectura y geología.

El propósito fundamental de un dibujo técnico es transmitir la forma y dimensiones exactas de un objeto. Un dibujo en perspectiva ordinario no aporta información acerca de detalles ocultos del objeto y no suele ajustarse a su proporción real. El dibujo técnico convencional utiliza dos o más proyecciones para representar un objeto. Estas proyecciones son diferentes vistas del objeto desde varios puntos que, si bien no son completas por separado, entre todas representan cada dimensión y detalle del objeto.

Hay una serie de normas establecidas para la realización de mapas topográficos, planos arquitectónicos y otros dibujos técnicos que se utilizan para proporcionar información a los posibles usuarios de los mismos. Cuando es posible, se dibujan los objetos a su tamaño real. Sin embargo, el dibujo de los pequeños detalles de máquinas se hace aumentando todas sus dimensiones de

forma proporcional, mientras que los mapas y planos de grandes instalaciones se realizan reduciendo sus dimensiones.

La mayor parte del dibujo técnico se realiza hoy con ordenadores, ya que es más fácil modificar un dibujo sobre la pantalla que sobre el papel. Los ordenadores también hacen más eficientes los procesos de diseño y fabricación. Por ejemplo, si las especificaciones de una pequeña pieza de una máquina se modifican en el ordenador, este puede calcular cómo afectan los cambios al resto de la máquina antes de proceder a su fabricación. Este tipo de dibujo asistido por ordenador se denomina CAD, y en el presente proyecto se ha utilizado un programa de este tipo para el diseño de la suspensión en 3D, *Solid Edge*.

La utilización de *Solid Edge* para la creación de las piezas de la simulación se debe a que el lenguaje VRML permite cargar las piezas diseñadas mediante un programa CAD. Además, el diseño de dichas piezas mediante la programación VRML es mucho más complicado.

### 3.1.1.- Gráficos orientados a objetos

Los gráficos orientados a objetos son gráficos de ordenador basados en el uso de elementos de construcción, como líneas, curvas, círculos y rectángulos. Los gráficos orientados a objetos, utilizados, por ejemplo, en el diseño asistido por ordenador y en programas de dibujo, describen un dibujo matemáticamente, como un conjunto de instrucciones que crean los elementos de la imagen. También existen los gráficos de mapas de bits, otro método muy extendido para crear imágenes, que representa los gráficos como un conjunto de puntos en blanco y negro o en color que siguen un patrón determinado.

Los gráficos orientados a objetos permiten al usuario manipular objetos como unidades completas, como, por ejemplo, al cambiar el largo de una línea o al aumentar el tamaño de un círculo, mientras que los gráficos de mapas de bits requieren pintar de nuevo cada uno de los puntos de una línea o círculo. Debido a que los objetos están descritos matemáticamente, los gráficos orientados a objetos se pueden estratificar, girar y ampliar con relativa facilidad.

El diseño asistido por ordenador, abreviado como DAO pero más conocido por las siglas inglesas CAD (*Computer Aided Design*), es el uso de un amplio rango de herramientas de programación que asisten a ingenieros, arquitectos y a otros profesionales del diseño en sus respectivas actividades.

El DAO es además la herramienta principal para la creación de entidades geométricas enmarcadas dentro de procesos de administración del ciclo de vida de productos (*Product Lifecycle Management*), y que involucra un *software* y, algunas veces, un *hardware* especiales. Los paquetes actuales varían desde aplicaciones basadas en vectores y sistemas de dibujo en 2 dimensiones (2D) hasta modeladores en 3 dimensiones (3D) a través del uso de modeladores de sólidos y superficies paramétricas.

Se trata básicamente de una base de datos de entidades geométricas (puntos, líneas, arcos, etc.) con la que se puede operar a través de una interfaz gráfica. Permite diseñar en dos o tres dimensiones mediante geometría alámbrica, esto es, puntos, líneas, arcos, superficies y sólidos para obtener un modelo numérico de un objeto o conjunto de ellos.

La base de datos asocia a cada entidad una serie de propiedades como color, capa, estilo de línea, nombre, definición geométrica, etc., que permiten manejar la información de forma lógica.

De los modelos pueden obtenerse planos con cotas y anotaciones para generar la documentación técnica específica de cada proyecto.

### **3.2.- *Solid Edge***

*Solid Edge* es un programa de diseño, con herramientas que permiten crear prototipos digitales en 3D. Con este programa, los proyectos llevan a una solución de diseño exacta, libre de errores. A través de sus herramientas de modelado y ensamblaje, se pueden desarrollar numerosos productos, desde piezas elementales, hasta conjuntos ensamblados [8].

*Solid Edge* ayuda a los profesionales del diseño mecánico a reducir los tiempos de comercialización gracias a la obtención de un diseño correcto desde el primer momento. *Solid Edge* contiene unas herramientas exclusivas para eliminar los errores de diseño y reducir así el tiempo de desarrollo y los costes.

Al diseñar con *Solid Edge*, no sólo se crean prototipos virtuales en 3D de los productos, sino que también se aprovecha el conocimiento con el que se consigue un proceso de ingeniería preciso y sin errores.

Sus principales características son las siguientes:

- Gestión de grandes conjuntos complejos formados por muchas piezas y subconjuntos.
- Gestión de datos de conjunto desde las primeras fases de planificación del proyecto hasta los ciclos de revisión, fabricación, mantenimiento del proyecto y archivado.
- Verificación visual del movimiento en un ensamble o verificación de interferencias en todo el rango de movimiento.
- Automatización y racionalización de todas las funciones de diseño, desde el concepto, hasta el diseño en detalle, y la producción de planos, para reducir notablemente el tiempo de desarrollo.

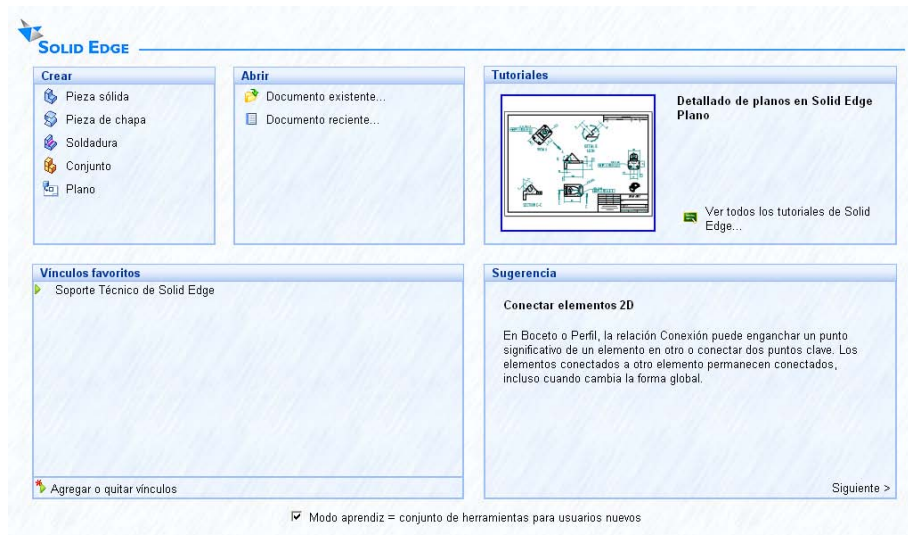
- Creación de modelos virtuales muy precisos que incorporan conocimientos de ingeniería para evitar errores costosos y trabajo innecesario.
- Ayuda para la evaluación de un mayor número de alternativas de diseño en menos tiempo, optimizando el rendimiento y aumentando la fiabilidad.
- En el diseño de ensamblajes, permite dividir las tareas de diseño entre los miembros del equipo, presentar los subensamblajes a medida que se terminan y garantizar un producto final homogéneo.
- Ayudas de diseño exclusivas para eliminar errores y facilitar la toma de decisiones.
- Simulación de desplazamientos complejos, detección de interferencias y creación de animaciones de la gama completa de movimientos de un ensamblaje con facilidad y precisión.
- Seguimiento de los dibujos de ingeniería, consecuencia de las alteraciones del modelo de diseño en 3D.
- Herramientas paramétricas basadas en operaciones para modelar piezas mecánicas.
- Controles de diseño, detalle, anotación, y dimensiones que se ajustan automáticamente al estándar del plano mecánico seleccionado.
- Sistema asociativo de producción de planos que crea y actualiza automáticamente los dibujos de modelos 3D.
- Herramientas económicas para compartir los datos de diseño y colaborar: *SmartView*, *WebPublisher*.
- Conversión de datos desde y hacia formatos CAD: ACIS, Autocad (DXF/DWG), IGES, MicroStation, STEP, Parasolid, Pro/ENGINEER, etc.
- Biblioteca de conjuntos y mecanismos que agregan las coordenadas de posicionamiento y las operaciones necesarias para la reutilización en nuevos diseños.
- Cálculo de masa y volumen, área de superficie y centro de gravedad.
- Importación en los formatos IGES, STEP, SAT, DWG, DXF, DGN, Pro-E, Parasolid, etc.
- Exportación en los formatos IGES, STEP, SAT, DWG, DXF, DGN, Pro-E, Parasolid, etc.

Las prestaciones descritas en los apartados anteriores hacen de *Solid Edge* una herramienta de gran utilidad en el diseño de:

- Utillaje y componentes.
- Maquinaria y estructura mecánica.
- Maquinaria general.
- Maquinaria de envasado y embalaje.
- Electromecánica.
- Maquinaria y equipo para la industria de proceso.
- Maquinaria agrícola, obras públicas y transporte.
- Maquinaria de corte, manipulación y mecanizado.

### 3.3.- Utilización de *Solid Edge*

Al ejecutar *Solid Edge* por primera vez aparece una ventana (Fig. 40) en la que se puede seleccionar el modo de inicio de la aplicación. Se puede crear un fichero nuevo tanto de pieza sólida, como de pieza de chapa, de conjunto, de plano o de soldadura. También se pueden ver los tutoriales, así como los accesos principales a las *Web* de *Solid Edge*. Hay un “modo aprendiz” que sirve de ayuda en la primera fase de iniciación con el programa.



**Fig. 40 – Ventana inicial de *Solid Edge***

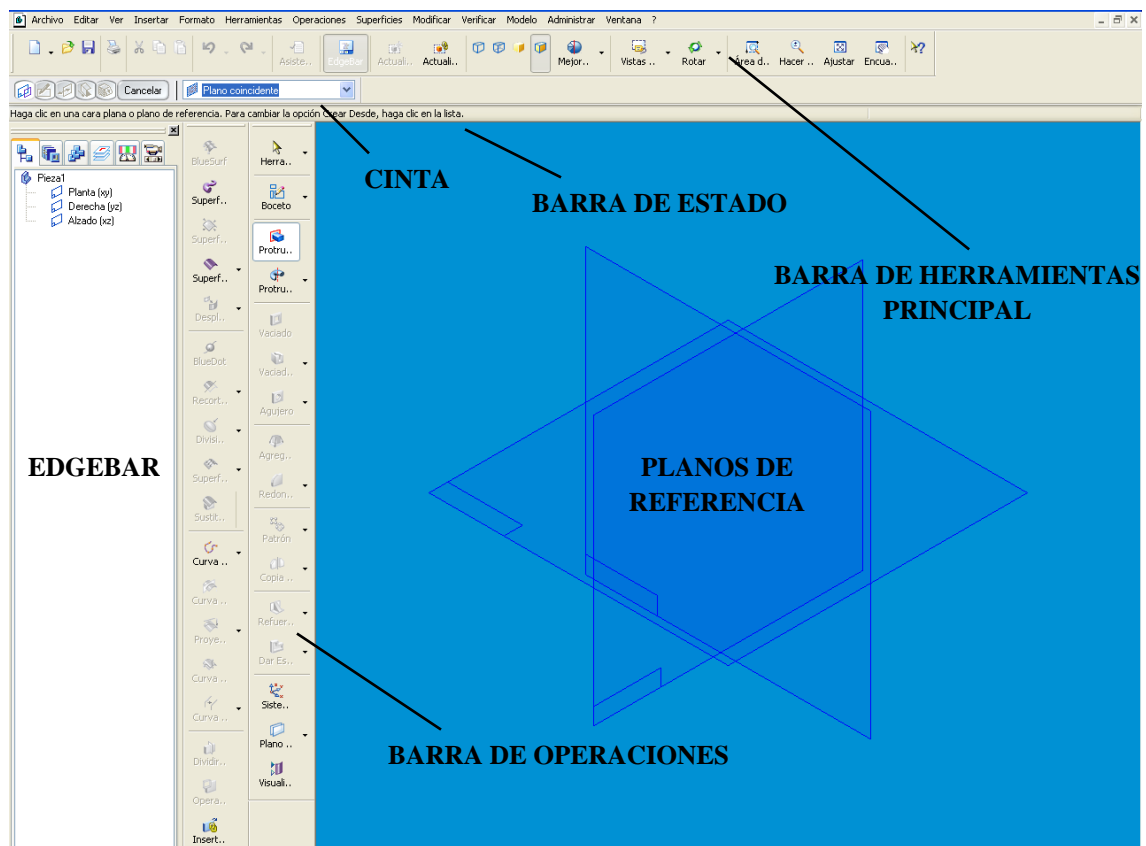
Si ya se ha iniciado *Solid Edge* y se desea abrir un nuevo archivo, sólo hay que ir al menú “Archivo – Nuevo” y seleccionar el tipo de archivo según el entorno deseado (Tabla 1).

PARA	SELECCIONAR
CONJUNTO	NORMAL.ASM
PIEZA SÓLIDA	NORMAL.PAR
PIEZA DE CHAPA	NORMAL.PSM
SOLDADURA	NORMAL.PWD
PLANO	NORMAL.DFT

**Tabla 1 – Tipos de archivo**

### 3.3.1.- Entorno

Al ejecutar *Solid Edge* – *Pieza sólida* se puede ver una ventana (Fig. 41) con los siguientes menús y barras de herramientas, que a continuación se comentan:



**Fig. 41 – Entorno de *Solid Edge***

### **- Barra de herramienta principal**

Es la barra de herramientas común en todo programa que trabaje bajo el sistema operativo Windows. En ella, se pueden ver las herramientas más comunes: Abrir, Guardar, Copiar, Imprimir, etc.

### **- Menú Cinta**

La cinta es una barra de herramientas dinámica sensible al contexto que aparece al hacer *clic* en un comando específico o al seleccionar un elemento.

### **- Barra de Operaciones**

La barra de operaciones muestra todas las operaciones que se pueden realizar sobre la pieza. Esta barra cambia en el modo de “Boceto” o en cualquier otro que así lo requiera.

### **- Edgebar**

Ventana de ayuda para la gestión de tareas. En ella van apareciendo las operaciones a medida que se van realizando.

### **- Planos de referencia (Excepto en el Entorno Plano)**

Permiten definir el plano del espacio sobre el que se va a trabajar.

### **- Barra de Estado**

Identifica la herramienta (operación), así como una pequeña ayuda sobre el proceso de utilización de la misma, cuando se pulsa o pasa el ratón sobre algún icono.

## **3.3.2- Diseño de las piezas de la suspensión**







A continuación, se describen las herramientas básicas de diseño que dispone *Solid Edge* para la creación de piezas, y se detalla el proceso del diseño de una de las piezas de este proyecto.

### 3.3.2.1.- Operaciones básicas

Las dos acciones básicas a la hora de crear piezas en *Solid Edge* son la protusión y el vaciado.



#### 3.3.2.1.1- Protusión

Consiste en crear un sólido tridimensional a partir de un boceto en dos dimensiones. Existen diferentes tipos de protusión, de las que se describirán las principales:






- **Protusión por Proyección**  : Extruye un perfil o boceto paralelamente al plano seleccionado.
- **Protusión por Revolución**  : Extruye el perfil seleccionado mediante una revolución alrededor de un eje.
- **Protusión por Barrido**  : Extruye un perfil a lo largo de una trayectoria. El perfil debe ser cerrado y la trayectoria continua y tangente en caso de que conste de varios elementos.
- **Protusión por Secciones**  : Crea una extrusión a través de varias secciones.
- **Protusión Helicoidal o Hélice**  : Construye una extrusión helicoidal con una sección transversal paralela o perpendicular al eje de la espiral.
- **Protusión Perpendicular**  : Construye una extrusión a 90°, respecto de una cara de una pieza, proyectando una curva cerrada o un boceto residentes en la cara. Para proyectar una curva cerrada se debe utilizar el comando “Proyectar curva” de la barra de herramientas “Superficies”.

#### 3.3.2.1.2.- Vaciado

Las operaciones de vaciado se tratan y actúan de la misma forma que las de protusión con la diferencia de que en lugar de crear material, lo eliminan. Los tipos de vaciado existentes en *Solid Edge* son los siguientes:

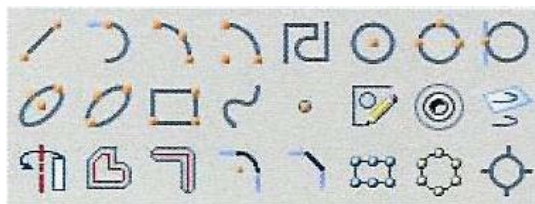
- **Vaciado**  : Vacía un perfil o boceto paralelamente al plano seleccionado.
- **Vaciado por Revolución**  : Vacía el perfil seleccionado mediante una revolución alrededor de un eje.



- **Vaciado por Barrido** : Vacía un perfil a lo largo de una trayectoria.
- **Vaciado por Secciones** : Vacía una extrusión a través de varias secciones.
- **Vaciado Helicoidal** : Vacía una extrusión de forma helicoidal, con una sección transversal paralela o perpendicular al eje de la espiral.
- **Vaciado Perpendicular** : Vacía una extrusión a 90°, respecto de una cara de una pieza, vaciando una curva cerrada o un boceto residentes en la cara.
- **Agujero** : Esta herramienta se utiliza en lugar de la herramienta “vaciado” cuando se quiere realizar agujeros roscados, abocardados o avellanados. La operación agujero circular del entorno boceto va ligada a esta operación, siendo esta herramienta la única que se podrá usar para realizar el perfil agujero.

#### 3.3.2.2.- *Proceso de diseño*

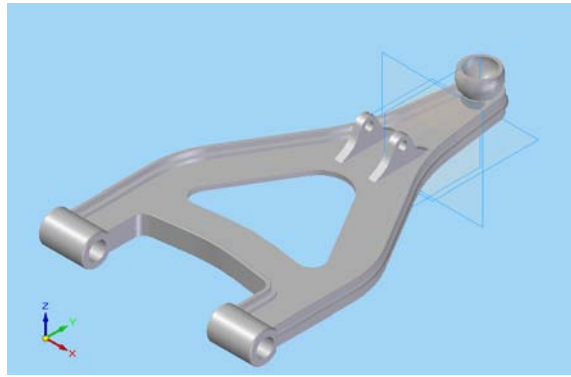
El primer paso a la hora de crear una pieza en *Solid Edge* es elegir una operación de protusión, ya que se necesita tener material sobre el que trabajar. Después, se debe elegir el plano sobre el que se quiere crear el boceto 2D que más adelante se transformará en 3D, por medio de la protusión seleccionada. Para crear un boceto se utilizan las herramientas de dibujo disponibles para 2D (Fig. 42).



**Fig. 42 – Herramientas para la creación del boceto**

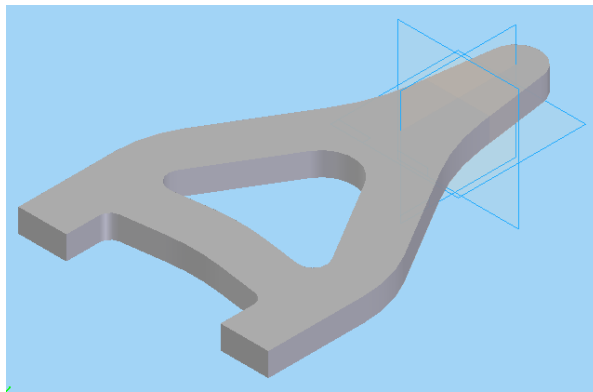
Con este grupo de herramientas se puede dibujar el boceto de la pieza que se quiere crear. Posteriormente, se crea el perfil y se consigue una pieza en 3D a la que se le irá dando la forma deseada, añadiendo o eliminado material por medio de más protusiones o vaciados.

A continuación, se describe el proceso de creación de la sujeción base de la suspensión doble *Wishbone*, objeto del presente proyecto, indicando cada uno de los pasos que se ha seguido para su diseño (Fig. 43).



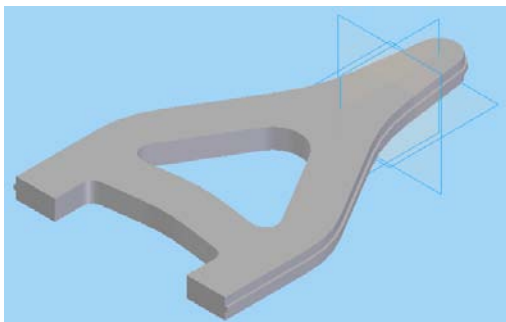
**Fig. 43 – Sujeción base**

En primer lugar se hace una protusión con la forma básica en “A” de la pieza, con sus medidas, y con el espesor total de la misma (Fig. 44).

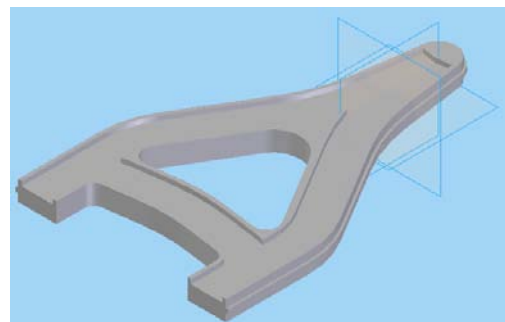


**Fig. 44 – Creación sujeción base I**

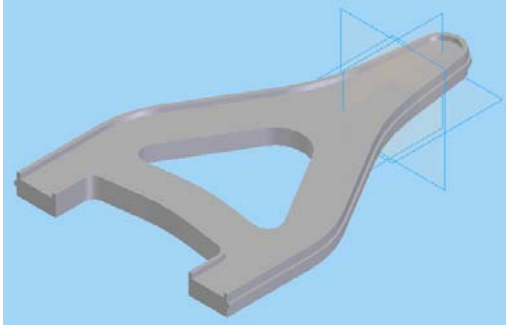
A continuación, se perfila el borde exterior de esta pieza mediante un vaciado por barrido de todo el perfil de la pieza, hacia el exterior (Fig. 45).



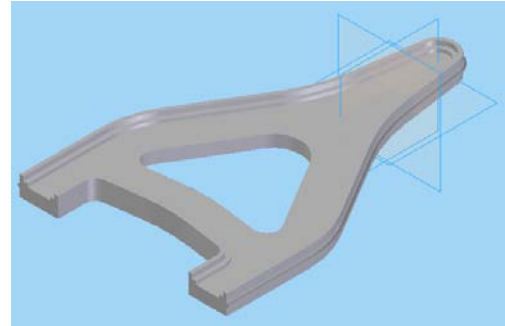
**Fig. 45 – Creación sujeción base II**



**Fig. 46 – Creación sujeción base III**



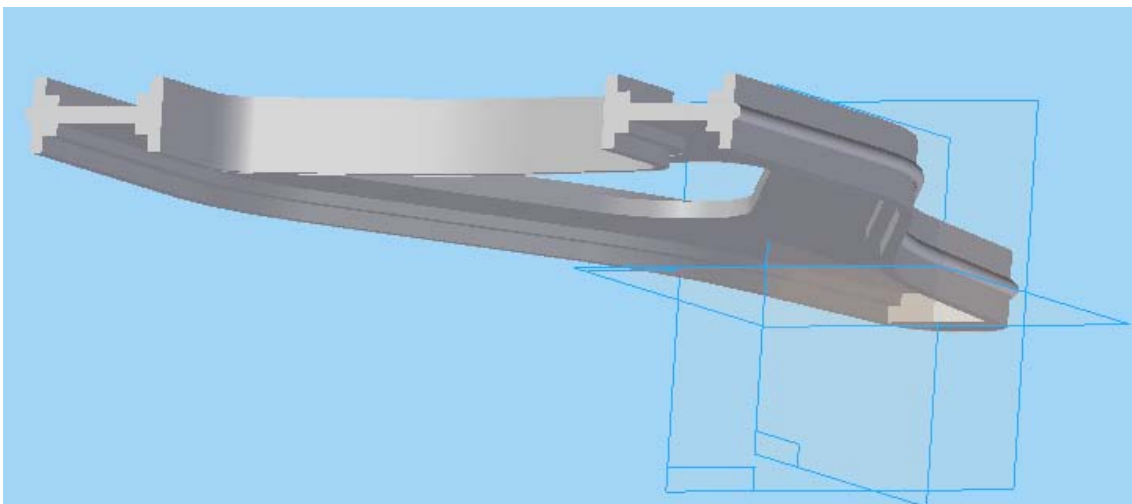
**Fig. 47 – Creación sujeción base IV**



**Fig. 48 – Creación sujeción base V**

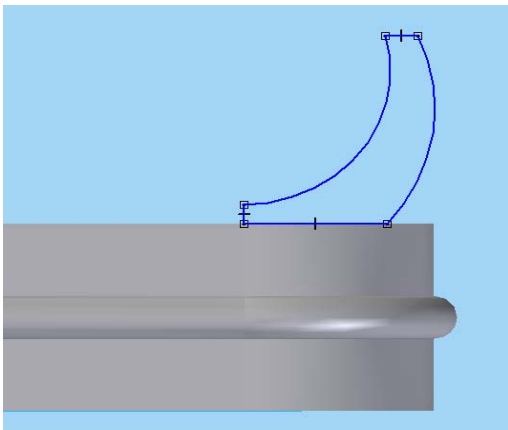
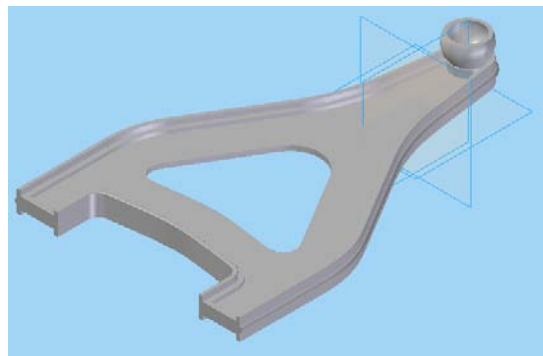
De la misma forma que se hace un barrido para perfilar el borde exterior de la pieza, se hacen una serie de vaciados por barrido por el interior del perfil de la misma, para conseguir hacer el rebajado de esta parte interior de la pieza (Fig. 46). Se hace un último vaciado de la superficie para eliminar el exceso de material que no habían eliminado los vaciados por barrido (Fig. 47), y se repite este proceso una segunda vez para realizar el segundo rebajado que tiene el perfil interior de la pieza. Por último, se redondea todo el perfil exterior. (Fig. 48).

Para conseguir el mismo perfil en la parte inferior de la pieza, se repiten los pasos que se han indicado en el párrafo anterior, quedando la pieza como se observa en la figura 49.

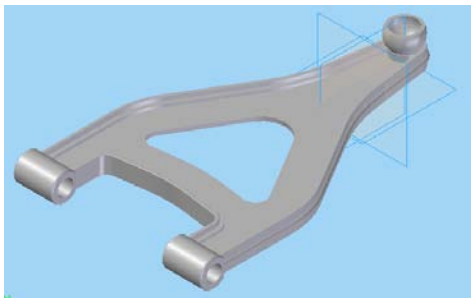
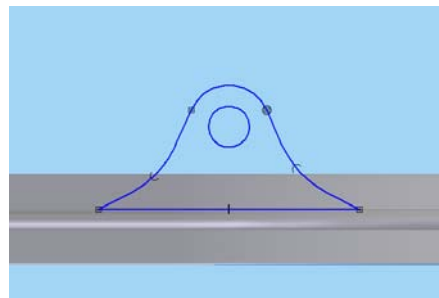


**Fig. 49 – Creación sujeción base VI**

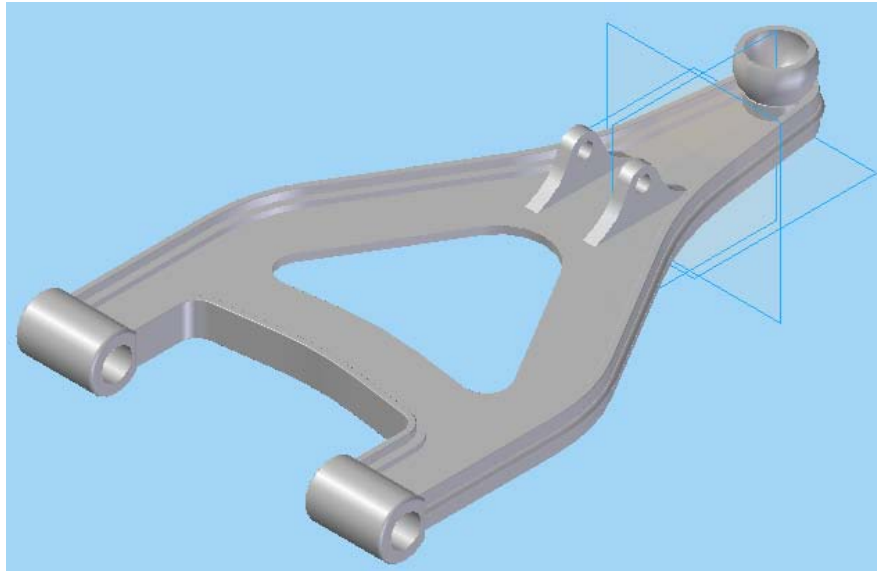
A continuación, se describe el proceso para la creación del hueco en el que reposa la rótula de la mangueta. Para ello, se crea una protusión por revolución de  $360^\circ$ , cuyo perfil se puede ver en la figura 50, alojado en el extremo más estrecho de la sujeción base, y cuyo eje de revolución se encuentra justo en el centro del radio del perfil de este extremo de la pieza. Después, se redondean todos los perfiles que tienen aristas vivas en la pieza. Por último, y mediante una protusión, se aumenta la superficie de los extremos de los “brazos” de esta pieza base, en el punto en el que se unen los perfiles que sirven de anclaje de la suspensión al chasis (Fig. 51).

**Fig. 50 – Creación sujeción base VII****Fig. 51 – Creación sujeción base VIII**

Para crear los perfiles de anclaje al chasis de los que se hablaba anteriormente (Fig. 52), se genera una protusión cilíndrica perpendicular a la pieza, desde uno de los extremos del brazo hasta el otro. A continuación, se realiza un vaciado recto de la parte central, eliminando el exceso de material en esta zona. Por último, se redondean todas las aristas vivas.

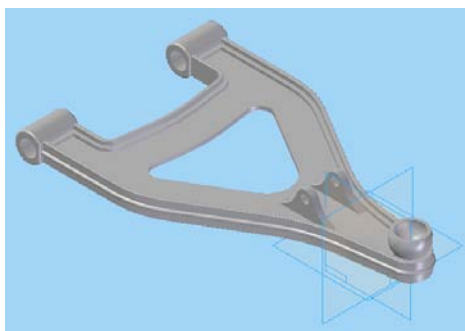
**Fig. 52 – Creación sujeción base IX****Fig. 53 – Creación sujeción base X**

Para terminar, se crea la sujeción para el tornillo donde va alojado el muelle y el amortiguador. Para ello, se crea una protusión con el perfil indicado en la figura 53 a la altura correspondiente, y después, se hace un vaciado transversal para eliminar el exceso de material. La figura 54 muestra la pieza final.

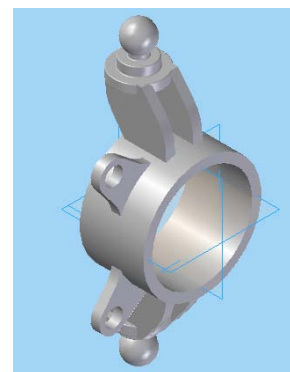


**Fig. 54 – Sujeción base**

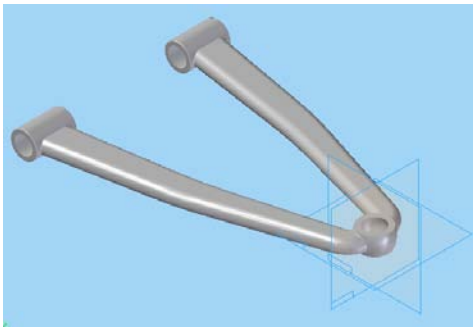
Finalmente, se muestran todas las piezas que se han creado en *Solid Edge* para la creación de la suspensión, y su posición definitiva para la creación posterior de los conjuntos (Figs. 55 a 64).



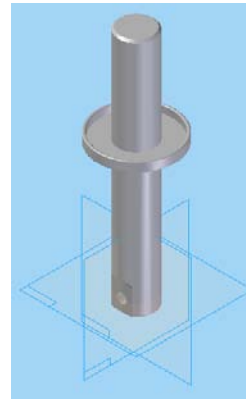
**Fig. 55 – Sujeción base**



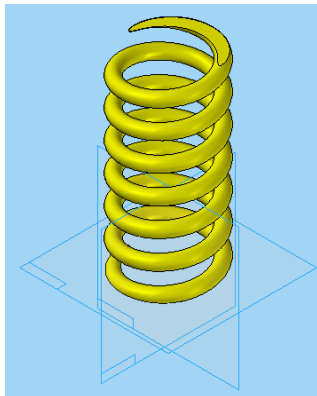
**Fig. 56 – Mangueta**



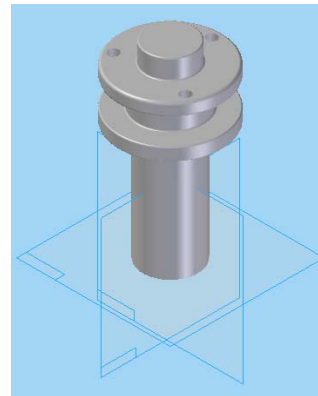
**Fig. 57 – Sujeción superior**



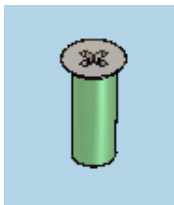
**Fig. 58 – Sujeción amortiguador**



**Fig. 59 – Muelle**



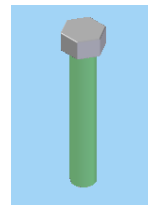
**Fig. 60 – Amortiguador**



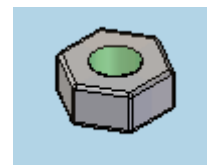
**Fig. 61 – Tornillo  
amortiguador**



**Fig. 62 – Tuerca  
amortiguador**



**Fig. 63 – Tornillo  
sujeción base**



**Fig. 64 – Tuerca  
sujeción base**

## **CAPITULO 4. LA REALIDAD VIRTUAL**

La introducción de las Nuevas Tecnologías de la información y la comunicación en todos los ámbitos de la vida, como es el caso de la Realidad Virtual (RV), está produciendo un cambio significativo en la manera de trabajar, de relacionarnos y de aprender. Estas tecnologías de la información se aplican al campo pedagógico con el objeto de racionalizar los procesos educativos, mejorar los resultados del sistema escolar y asegurar el acceso al mismo de grupos convencionalmente excluidos [7].

“Realidad Virtual” es un término muy común. En estos días en que todo es virtual, se encuentra mucha confusión en las personas expuestas de una u otra forma a los nuevos medios. En el nombre en sí hay una gran contradicción: Realidad Virtual. Algo que es, pero no es. La realidad virtual es una representación de las cosas a través de medios electrónicos, que dan la sensación de estar en una situación real en la que se puede interactuar con lo que se encuentra alrededor de la persona que la está experimentando [11].

Su función es la de crear un mundo con objetos, definir las relaciones entre ellos y la naturaleza de las interacciones entre los mismos. La realidad virtual permite observar un objeto o estar dentro de él, es decir, penetrar en ese mundo que sólo existe en la memoria del observador un corto plazo de tiempo (mientras lo observa) y en la memoria del ordenador.

Hoy en día existen muchas aplicaciones de entornos de realidad virtual con éxito en muchos de los casos. En estos entornos el individuo sólo debe

preocuparse por actuar, ya que el espacio que antes se debía imaginar, es facilitado por medios tecnológicos.

#### 4.1.- Historia

La expresión “mundos virtuales” fue usada por primera vez por el consultor en administración Donald Schon en 1982, para referirse a las imágenes que mantiene un sujeto acerca del entorno con el cual interactúa. Posteriormente, estos términos han sido retomados, discutidos y difundidos por el Dpto. de Ciencias de la Computación de la Universidad de Carolina del Norte.

El primer sistema que se proponía atravesar el umbral de las dos dimensiones y recrear una verdadera “telepresencia en un espacio de datos” (*Telepresence in Dataspace*) fue concebido en los laboratorios Ames de la NASA en California. El proyecto *Virtual Environment Display*, iniciado en 1984, tenía como objetivo la construcción de una estación de trabajo virtual destinada a ser utilizada en misiones espaciales de la NASA.

En el año 1994, durante la primera conferencia anual de *World Wide Web* en Génova, Berners-lee y Regett, organizaron una reunión, para discutir acerca de las interfaces de la RV. Después de esta reunión se llegó a la conclusión de que muchos participantes estaban trabajando en proyectos para conseguir herramientas de visualización tridimensional que interoperaran con Internet. El mayor esfuerzo se concentró en torno a una lista de correo donde se llegó a un consenso sobre los requerimientos para conseguir un lenguaje modelado de Realidad Virtual, y así discutieron acerca de las tecnologías existentes. Después de varias propuestas, se escogió la sintaxis de *OpenInventor* de Silicon Graphics como base de un formato de descripción de objetos geométricos texturizados, agregando la posibilidad de combinar objetos guardados remotamente en la red. De esta manera nació VRML 1.0.

Este lenguaje buscaba tres objetivos:

1. Independencia de plataforma.
2. Extensibilidad.
3. Habilidad para trabajar adecuadamente con conexiones de poco ancho de banda.



## 4.2.- Características de un sistema de realidad virtual

Las características más importantes de un sistema de realidad virtual son las siguientes:

- **Inmersión:** propiedad mediante la cual el usuario tiene la sensación de encontrarse dentro de un mundo tridimensional.
- **Existencia de un punto de observación o referencia:** permite determinar la ubicación y la posición de observación del usuario dentro del mundo virtual.
- **Navegación:** propiedad que permite al usuario cambiar su posición de observación.
- **Manipulación:** característica que posibilita la interacción y transformación del entorno virtual.
- Responde a la metáfora de “mundo” que contiene “objetos” y opera en base a reglas que varían en flexibilidad dependiendo de su compromiso con la Inteligencia Artificial.
- Se expresa en lenguaje gráfico tridimensional.
- Su comportamiento es dinámico y opera en tiempo real.

### 4.2.1.- Tipos de realidad virtual

La realidad virtual puede ser de dos tipos: inmersiva y no inmersiva [11]. Los métodos inmersivos de realidad virtual con frecuencia se ligán a un ambiente tridimensional creado por ordenador, el cual, se manipula a través de cascos, guantes u otros dispositivos que capturan la posición y rotación de diferentes partes del cuerpo humano. La realidad virtual no inmersiva utiliza medios como el que actualmente ofrece Internet, en el cual se pueden interactuar en tiempo real con diferentes personas, en espacios y ambientes que en realidad no existen, sin la necesidad de dispositivos adicionales al ordenador.

#### 4.2.1.1.- Realidad virtual no inmersiva

La realidad virtual no inmersiva ofrece un nuevo mundo a través de una ventana de escritorio. Este enfoque no inmersivo tiene varias ventajas sobre el enfoque inmersivo como el bajo coste y la fácil y rápida aceptación de los usuarios. Los dispositivos inmersivos son de alto coste y generalmente el usuario

prefiere manipular el ambiente virtual por medio de dispositivos familiares como son el teclado y el ratón, que por medios de cascos pesados o guantes.

Actualmente Internet proporciona medios para que diferentes personas puedan reunirse en el mismo espacio virtual. En este sentido, Internet tiende a ser un mecanismo de telepresencia. Este medio brinda espacios y realidades que físicamente no existen, pero que, sin embargo, forman parte de nuestras formas de vida.

#### *4.2.1.2.- Realidad virtual inmersiva*

Los métodos inmersivos de realidad virtual (Fig. 65) se ligan con frecuencia a un ambiente tridimensional creado por un ordenador, el cual se manipula a través de cascos, guantes u otros dispositivos, que capturan la posición y rotación de diferentes partes del cuerpo humano, y por medio de los cuales, los estudiantes pueden llegar a "sentir" la textura, dimensiones e inclusive la temperatura de objetos virtuales que existen dentro del mundo virtual.



**Fig. 65 – Realidad virtual inmersiva**

### **4.3.- Equipos utilizados para realidad virtual**

Existen diferentes equipos utilizados en la realidad virtual y se pueden incluir en diferentes categorías, siendo las principales las indicadas a continuación.

#### 4.3.1.- Para visión

La realidad virtual en el área de la visión trabaja básicamente con dos tipos de implementos: cascos y *boom*. Este último es un equipo que consiste en un brazo mecánico que sostiene un display a través del cual al girarlo se puede observar el entorno del mundo virtual en el cual se está. Debido a que su peso es soportado por el brazo mecánico y no por el usuario, como ocurre con el casco, este puede ser un equipo de mayor complejidad y contenido electrónico. Lo cual se traduce en ventajas tales como la obtención de una mejor resolución. A continuación se presentan algunas características de estos equipos:

- **Visión estereoscópica:** Es la sensación de ver una determinada imagen en 3 dimensiones. Esto se logra haciendo una representación igual para cada ojo de la imagen que se va a observar. Estas representaciones son posteriormente proyectadas desde un mismo plano y separadas una distancia que está determinada por la distancia a la cual se encuentra el observador del plano de las imágenes. Desde este punto de vista, también existen equipos de visión monocular a través de los cuales se visualizan los objetos en la forma habitual.
- **Binoculares:** Son equipos que constan de una pantalla individual para cada ojo. Para el funcionamiento de la visión estereoscópica es necesario tener un equipo que tenga esta característica. Así mismo, también existen equipos monoculares, los cuales constan de una sola pantalla para ambos ojos.

#### 4.3.2.- Para interactuar

En la actualidad, la realidad virtual está haciendo uso de los guantes y vestidos como medio para interactuar en un ambiente virtual. Para lograr esto, estos dispositivos se comportan inicialmente como dispositivos de entrada que le permiten al ordenador conocer la ubicación del usuario dentro del ambiente virtual. Así mismo, le permiten al usuario ubicarse en el medio e interactuar con él y, en algunos casos, recibir ciertos estímulos donde estos dispositivos se convierten en dispositivos de salida. Algunas sensaciones o estímulos que se pueden recibir son:

- Sensación de estar sosteniendo un objeto que se ha cogido dentro del ambiente virtual; esto se logra gracias a unas almohadillas que se inflan en el guante y dan la sensación de percibir un peso.
- También se puede llegar a percibir la rugosidad y la forma propias de objetos situados en el interior del ambiente virtual, lo cual se logra gracias a que algunos dispositivos tienen partes de aleaciones con

memoria que tras variaciones de temperatura toman formas que se les han introducido con anterioridad.

Los denominados “dispositivos de seguimiento” son aquellos mediante los cuales el ordenador localiza al usuario dentro del ambiente virtual. Uno de los más utilizados es el *tracking* óptico, mecanismo que consta de un casco que es llevado por la persona que se encuentra inmersa en la escena virtual. Este casco tiene en su parte superior una cámara que enfoca el techo de la sala dentro de la cual se desplaza el usuario. En el techo se encuentran ubicadas unas lámparas que se encienden y apagan de forma secuencial a gran velocidad y las cuales reciben la señal enviada por la cámara. La lámpara que se enciende en el momento en que la persona pasa bajo ella es la que envía la señal de ubicación al ordenador.

#### 4.3.3.- Para audición

Los audífonos son el equipo básico para escuchar los sonidos propios de un ambiente virtual. A continuación se presentan algunas variantes de estos equipos:

- **Audífonos convencionales:** Son los audífonos de uso más corriente. A través de estos se escucha el sonido simulado de los objetos sin identificar auditivamente el punto de ubicación del mismo.
- **Convolvotrón:** Estos audífonos, además de simular el sonido propio de los objetos, simulan la ubicación de los mismos dentro del ambiente virtual.
- **Cabina de Simulación:** Como su nombre indica, son grandes aparatos en los que se sumerge al usuario como si maniobrara un vehículo con cabina (simuladores de vuelo, submarinos, etc.).
- **Holofonía:** Este sistema de sonido funciona por medio de unos audífonos convencionales. A través de ellos, se consigue simular la posición y la distancia de los diferentes sonidos de forma muy realista.

#### **4.4.- Ejemplos de aplicación de la realidad virtual**

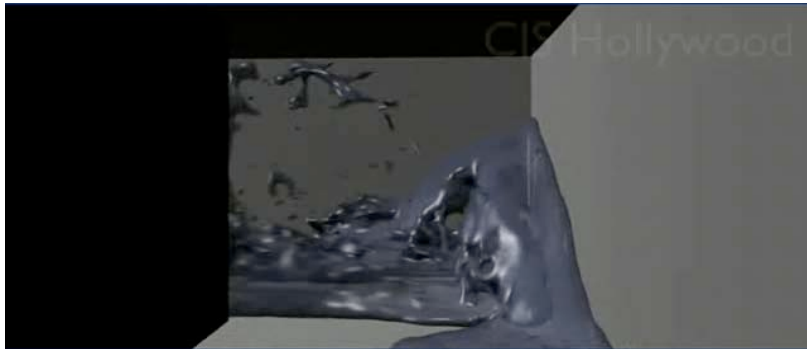
En un principio, la realidad virtual fue usada en su mayoría para aplicaciones militares o incluso de entretenimiento. Sin embargo, en los últimos años se han diversificado las áreas en las que se utiliza. En las secciones anteriores, se mencionó los diferentes tipos de realidad virtual y sus áreas de utilización. En este punto, se explorarán más a fondo los distintos proyectos que

existen relacionados con esta tecnología. Se describirán proyectos de distintos tipos: visualización, manipulación de robots o medicina, entre otros [14].

#### 4.4.1.- Realidad virtual en la física

Dentro del área de la física existen proyectos con distintos enfoques. Aquí se describe una aplicación muy común: la visualización de fluidos de partículas (Fig. 66).

Existen proyectos que modelan este tipo de fenómeno, donde el propósito principal es el fácil análisis de una gran cantidad de datos que facilitan el estudio de los modelos. Se cuenta con una herramienta auxiliar que permite visualizar modelos complicados de interpretar si sólo se analizan tal cual. Este tipo de proyecto corresponde a un tipo de realidad no inmersiva.



**Fig. 66 – Fluido de partículas**

#### 4.4.2.- Realidad virtual en la ingeniería

Dentro de las áreas de ingeniería existen investigaciones referentes a los procesos de ensamblado. También existen áreas dedicadas al desarrollo de prototipos virtuales. Todas estas aplicaciones facilitan la automatización dentro de diferentes áreas.

##### *4.4.2.1 - Proceso de ensamblado*

Cuando se tiene un proceso de ensamblado de algún producto, se presentan distintos acontecimientos como las deformaciones de plástico, fricción externa, fenómeno termal, absorción; y factores como el desgaste de herramientas, ocasionando errores de dimensión y forma. Si se tiene información adicional sobre el efecto de los parámetros antes mencionados sobre la variación en los

valores de tolerancia y dimensión, se puede desarrollar mecanismos para el ensamblado automático.

Usando un modelo de elementos finitos se puede visualizar las fuerzas que actúan en el proceso de manufactura y la deformación del equipo bajo la acción de estas fuerzas. Si se tiene un ingeniero en diseño y manufacturación que pueda observar el ensamblado de una de las partes por medio del ordenador y dispositivos especiales, puede sugerir cambios en la tolerancia de los valores basándose en las condiciones de las máquinas, herramientas, fisuras y requerimientos de diseño. Un tipo de aplicación como ésta puede permitir obtener una configuración de ensamblado óptimo para satisfacer los requerimientos funcionales. Por lo que, es un tipo de herramienta efectiva para el proceso de toma de decisiones. Este tipo de proyectos son totalmente inmersivos.

#### *4.4.2.2 - Prototipos virtuales*

El diseño de prototipos es combinado con un modelado virtual de estos permitiendo al diseñador ser una parte muy importante en el detalle del diseño y la optimización del proceso.

Las técnicas de realidad virtual permiten generar ambientes computarizados para que el diseñador investigue y pruebe múltiples cambios a los diseños que está realizando mientras observa y manipula objetos virtuales al usar movimientos humanos naturales. Los diseños interactivos permiten cambiar los parámetros de diseño e inmediatamente determinar el efecto de los cambios.

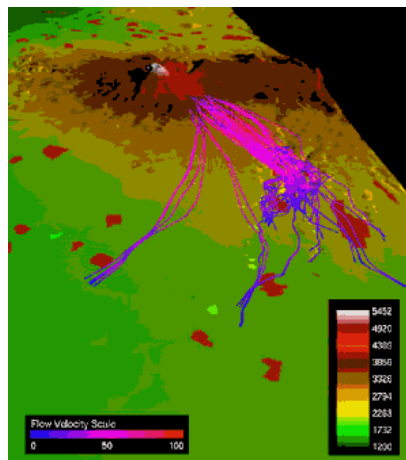
Existen compañías que utilizan la RV como una herramienta en el diseño de dispositivos de control de polución y de calderas. De esta forma, la compañía puede garantizar el funcionamiento de sus productos incluso antes de haberlos construido. Trabajando con las especificaciones de los productos, se modelan nuevas calderas y se simula su temperatura, dirección y velocidad de consumo de los gases. De esta forma, al realizar distintos experimentos con la colocación de los inyectores y otras características físicas, se crea el mejor sistema controlador de polución para la caldera y se integra dentro del diseño antes de que la caldera sea construida.

#### 4.4.3.- Realidad virtual en ciencias de la tierra

Dentro del área de Ciencias de la Tierra se realizan proyectos para algunas de las áreas de aplicación como lo es la visualización de fenómenos volcánicos o la modelación de relieves topográficos.

##### *4.4.3.1 - Visualización de fenómenos volcánicos*

Las simulaciones de fenómenos volcánicos (Fig. 67) permiten analizar la pérdida de vida y la destrucción de las infraestructuras. Los modelos de flujos permiten estimar los movimientos de materiales volcánicos dentro y sobre la superficie. Este tipo de aplicaciones permite el entendimiento de los peligros de estos fenómenos antes de que sucedan. Además del desarrollo de mapas de riesgo, asistencia en crisis y reconstrucción post-crisis. La Universidad de Buffalo, desarrolló un sistema de flujos que van desde lava en movimiento lento y flujos saturados (lo que les permitirá a oficiales públicos, científicos y a la población en general a entender el efecto de los fenómenos volcánicos), hasta diseñar planes apropiados de migración.

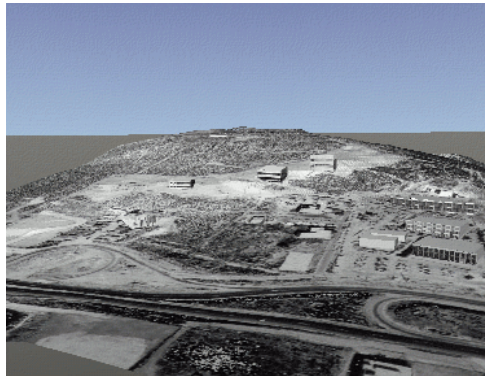


**Fig. 67 – Simulación de una erupción volcánica [14]**

##### *4.4.3.2 - Modelados de relieves topográficos*

El modelado de relieves topográficos (Fig. 68) es una herramienta que, aunada a las técnicas ya utilizadas, ayuda en el análisis de terrenos. Algunas Universidades han incorporado el modelado virtual de relieve topográfico de sus Campus a un Proyecto Ecológico de Crecimiento Armónico, donde además de contar con fotos de las distintas divisiones del campus, se tiene el modelo virtual

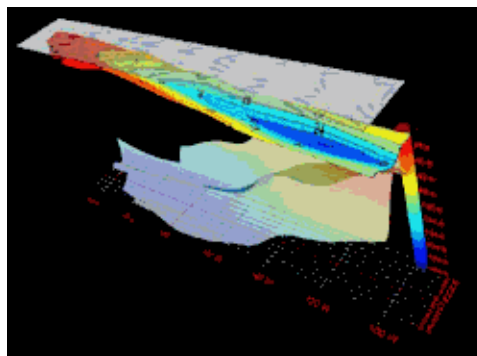
como una ayuda para la fácil planificación y localización de los cambios que se incorporen al campus.



**Fig. 68 – Modelado de relieves topográficos [14]**

#### 4.4.4.- Realidad virtual en la oceanología

Utilizando la realidad virtual en proyectos de oceanología (Fig. 69), se puede visualizar una estructura tridimensional de la superficie del océano, donde se puede modelar, por ejemplo, el comportamiento de las mareas, tener una simulación de cómo el viento afecta a las olas, u observar fenómenos como los del “El Niño” o “La Niña”, observando temperaturas, direcciones de vientos o velocidad.



**Fig. 69 – Simulación de la superficie del océano [14]**

#### 4.4.5.- Realidad virtual en la medicina

La medicina es uno de los campos más importantes para las aplicaciones de RV. Así como la medicina cuenta con una vasta diversidad de áreas de estudio, de igual manera la RV se aplica para diversas disciplinas (Fig. 70). A continuación se presentan algunos tipos de proyectos dentro de este área. Estas aplicaciones corresponden al tipo de realidad virtual inmersiva.



#### 4.4.5.1 - Realidad virtual en la Psicología

El tratamiento de fobias juega un lugar muy importante dentro de la disciplina de la psicología. Cada año, millones de personas acuden a psicólogos para el tratamiento de fobias, entre los que se encuentran el miedo a las alturas, a volar, a las arañas, claustrofobia, miedo a conducir, etc. La incorporación de la realidad virtual a los tratamientos ya establecidos beneficia y agiliza en gran parte el proceso de superación, puesto que el tener modelos virtuales (como, por ejemplo, modelos de elevadores ó simulación de vuelos, entre otros) aumenta la confianza y seguridad del paciente, puesto que él sabe que, mientras pruebe este tipo de modelos, nunca se encontrará realmente en peligro. Además, el paciente se siente más tranquilo al saber que al encontrarse en un ambiente virtual puede parar cuando lo desee. También puede visualizar mejor el problema y, en consecuencia, el psiquiatra puede observar lo mismo que el paciente está percibiendo, llevando un tratamiento más enfocado.

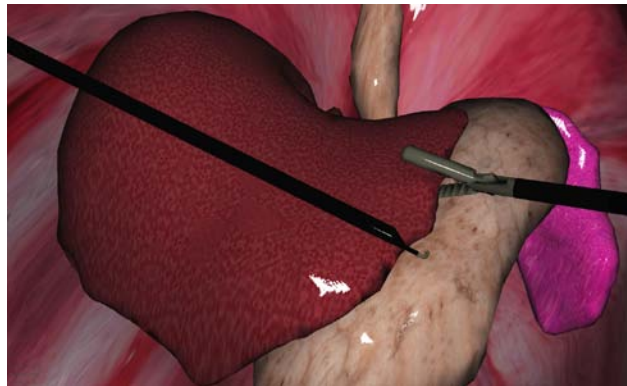


**Fig. 70 – Realidad virtual en procesos de rehabilitación [15]**

Este tipo de aplicaciones muestran un futuro prometedor para este tipo de tratamientos:

- Tratamiento para el bienestar de pacientes con cáncer: a través de la RV, se pueden experimentar mundos virtuales sin que se les cause fatiga alguna o estrés adicional.
- Simulación de cirugías virtuales (Fig. 71): En la Universidad de Washington se está experimentando con clases demostrativas de cirugía virtual. En esta universidad se ha creado un "cadáver virtual", donde los estudiantes pueden empuñar un bisturí virtual y practicar. En este sentido, es fácil imaginar un mundo virtual creado, por ejemplo, con VRML, que represente un completo quirófano virtual

internacional, en el que se recogieran las mejores técnicas quirúrgicas de distintos médicos de cualquier parte del mundo. Esta información puede servir de aprendizaje para los estudiantes de medicina y también para otros médicos [10].



**Fig. 71 – Realidad virtual en la cirugía [15]**

#### 4.4.6.- Realidad virtual en el entretenimiento

Cada día aumentan más las opciones de entretenimiento que utilizan la realidad virtual. Cada vez se hacen más películas en 3D, y de mayor calidad, lo que provoca que el espectador se meta y disfrute aún más de la película.

También cabe destacar la industria de los videojuegos (Fig. 72). Ya se están desarrollando equipos capaces de hacer que sea el propio jugador el que, con su propio movimiento, sea el que dirija al protagonista del videojuego.



**Fig. 72 – Realidad virtual en los videojuegos**

#### 4.4.7.- Realidad virtual en el arte

El mundo del arte es, y siempre ha sido, importante para el ser humano. Hoy en día en que la informática tiene influencia en casi cualquier área de trabajo o de interés, la RV también se ha convertido en una herramienta esencial para el campo de las artes. A continuación se describen algunas de las áreas de mayor influencia.

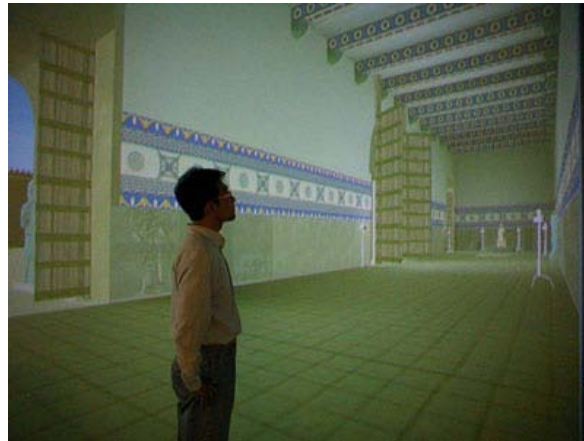
##### *4.4.7.1 - Realidad virtual en museos y planetarios*

La realidad virtual juega un papel importante para el conocimiento. Es utilizado por museos, planetarios y centros de ciencia (Fig. 73). Estos centros realizan exposiciones virtuales donde se pueden hacer recorridos por templos antiguos, palacios, galaxias, aprender de diversas áreas de conocimiento, entre otras.

En algunos de los proyectos realizados por estos centros, se experimenta con situaciones más cotidianas o con las que los visitantes (principalmente jóvenes y niños) pueden identificarse. Por ejemplo, se puede diseñar una montaña rusa y posteriormente experimentar el viaje como si físicamente se estuviese en la montaña.

Otro de los enfoques que se da a la RV es experimentar visitas virtuales a lugares o templos antiguos que, por alguna razón, no están disponibles al usuario (destrucción, restauración).

El lenguaje VRML está permitiendo ofrecer en Internet versiones virtuales de cualquier tipo de museo o galería de arte del mundo. De esta forma, cualquier estudiante puede acceder, no sólo a la imagen digitalizada de un cuadro y a explicaciones textuales, sonoras o audiovisuales sobre el mismo, sino también conocer las instalaciones del museo y recorrerlas virtualmente [10].



**Fig. 73 – Visita virtual de un templo [15]**

#### *4.4.7.2 - Realidad virtual en la arquitectura*

La manera que los arquitectos comunican sus ideas la mayor parte de las veces es de forma visual. Utilizar alguna forma de visualización, facilita la comprensión de información compleja, así como la comunicación. Hoy en día, cada vez son más los arquitectos que utilizan la RV como una herramienta más para hacer participar a los demás de sus ideas y trabajos (Fig. 74).

Algunos de los enfoques más comunes que los arquitectos dan al uso de la RV es en el modelado virtual de sus diseños de casas y edificios, donde además de hacer los diseños tradicionales como planos y maquetas, elaboran un modelo tridimensional interactivo, donde sus clientes pueden contemplar de una manera más “real” los diseños e inclusive, adentrarse en estos edificios o casas y recorrerlos libremente, teniendo así una visión más clara de las ideas que se tratan de expresar.

Además, existe un vínculo entre la arquitectura como tal y diseñadores urbanos, donde, no sólo se realizan los diseños de una casa o edificio, sino de un planteamiento más amplio como es el diseño de una ciudad o parte de ella. En este tipo de proyectos, la visualización va un poco más lejos. Se trata de planear con anticipación el crecimiento de una ciudad o una parte de ella.



**Fig. 74 – Edificio virtual creado con VRML [10]**

#### 4.4.8.- Realidad virtual en la enseñanza

Para que los medios tecnológicos queden integrados en el trabajo cotidiano de las aulas, se requiere la participación del profesor. Es él quien, en cada situación de aprendizaje, con sus decisiones y su actuación, conseguirá que el medio quede integrado. El profesor pasará de ser el elemento exclusivo en la transmisión de conocimientos, a convertirse en una pieza clave del proceso enseñanza-aprendizaje, como elemento mediador, generador y organizador de las situaciones de aprendizaje [7].

Según afirma García Ruiz (1998), a partir de los experimentos llevados a cabo por Sherman y Judkins (1994) en la Universidad de Washington se puede llegar a la conclusión de que con la realidad virtual los estudiantes "pueden aprender de manera más rápida y asimilar información de una manera más consistente que por medio del uso de herramientas de enseñanza tradicionales (pizarra, libros, etc.), ya que utilizan casi todos sus sentidos. Los estudiantes no sólo pueden leer textos y ver imágenes dentro de un casco de Realidad Virtual, sino que además pueden escuchar narraciones, efectos de sonido y música relacionados con el tema que están aprendiendo" [10].

Uno de los tradicionales problemas de la aplicación de la Realidad Virtual en la enseñanza es que, debido a su elevado precio, esta tecnología no está al alcance de los estudiantes y profesores. Precisamente la aparición del lenguaje VRML (*Virtual Reality Modeling Language*) ha paliado en cierta medida este inconveniente, haciéndola asequible a cualquier persona que posea simplemente un ordenador y un navegador de Internet. Obviamente, sólo con estos dispositivos se pierde el sentido del tacto al carecer de guantes, pero la sensación de inmersión en un mundo virtual sigue siendo la misma [10].

#### **4.5.- Futuro de la realidad virtual**

Uno de los objetivos de la RV es la creación del ciberespacio, en su concepción más imaginativa, plasmada a través de novelas. Algunos de los requisitos fundamentales de este ciberespacio es que sea gráfico, multiparticipativo, distribuido e independiente de plataforma.

Para lograr la creación de este ciberespacio, es necesario sobrepasar varios problemas actuales tales como el diseño gráfico (especialmente en los PCs, ya que son los que posee la mayoría de la población), la latencia y la velocidad de red, y la creación de un modelo de interacción con miles de participantes.

El siguiente paso importante hoy en día para la RV es la creación de un marco que permita comportamientos, entendidos estos como un cambio en el mundo tridimensional a través del tiempo, y la posibilidad del usuario de causar o ser afectado por dichos cambios. Dichos cambios podrían ser activados por interacción del usuario, el paso del tiempo, y otros objetos. Por simplicidad de diseño, los comportamientos se han clasificado en simples (un usuario con su ambiente) y en complejos (multiusuario).

Los ambientes virtuales pueden representar cualquier mundo tridimensional que puede ser real o abstracto. Esto incluye tanto sistemas reales como edificios, aeronaves, sitios de excavación, anatomía humana, reconstrucción de crímenes, sistemas solares, etc., como sistemas abstractos se pueden incluir campos magnéticos, modelos moleculares, sistemas matemáticos, acústica de auditores, densidad de población, etc. Estos mundos virtuales pueden ser animados, interactivos, compartidos y pueden exponer comportamiento y funcionalidad.

## **CAPITULO 5.- EL LENGUAJE VRML**

VRML (*Virtual Reality Modeling Language*) es un lenguaje de programación. Aunque presenta similitudes con otros lenguajes como BASIC o “C”, el VRML ha sido diseñado de forma específica para manejar gráficos, y contiene varias características integradas que facilitan la creación de modelos detallados.

El lenguaje VRML es ahora un poderoso lenguaje gráfico. Se trata de la adaptación de un lenguaje desarrollado por la compañía *Silicon Graphics*, cuyo predecesor fue *Open Inventor*, un sistema desarrollado con el fin de crear un lenguaje gráfico con múltiples capacidades [19].

*Open Inventor* fue diseñado para que los programadores pudieran elaborar ambientes realistas en tercera dimensión con un mínimo de conocimientos sobre programación gráfica, pudiendo aprovechar sus características avanzadas. Los desarrolladores de *Open Inventor* tomaron los mejores componentes de los lenguajes gráficos existentes en aquella época y los sintetizaron en un sólo conjunto que, a pesar de tener una gran capacidad expresiva y flexibilidad, resultaba fácil de aprender y entender.

Sin embargo, *Open Inventor* no es idéntico al VRML. Este último cuenta con ciertas características que lo hacen compatible con Internet y ha mejorado algunas de las opciones originadas con *Open Inventor*, pero aún así necesitaba mayor flexibilidad. Es por eso que Gavin Bell, uno de los desarrolladores de *Open Inventor SGI*, es también uno de los principales Arquitectos del VRML. Gavin volcó su experiencia en el diseño del VRML y en las modificaciones de su

antecesor, al eliminar algunas características poco útiles y agregar otros factores necesarios para el manejo de Internet.

El VRML es un lenguaje para descripción de escenas, no tanto un lenguaje de programación, puesto que no funciona de la misma manera que algunos lenguajes de programación como “C”, donde hay que compilar el programa y después ejecutarlo [19].

El sistema VRML pasa por un análisis sintáctico antes de ser desplegado en pantalla. La descripción de escenas es un proceso estático, pues los elementos no cambian cuando el archivo VRML se carga. De hecho, es posible cambiar el punto de vista, pero no la escena en sí.

La especificación original de este lenguaje, denominada VRML 1.0, data de Octubre de 1994, y se basaba en un producto de la Compañía *Silicon Graphics*. En 1994 se creó el *VRML Architecture Group* (VAG), con el objetivo de ayudar en la clarificación e implementación de la especificación inicial de este nuevo lenguaje. Con posterioridad, este organismo ha sido sustituido por el *VRML Consortium*, entre cuyos miembros se encuentra *Netscape*, *Microsoft*, *IBM* o *Silicon Graphics* [12].

## 5.1.- Versiones de VRML

Hasta el momento existen dos versiones de VRML:

### 1.- VRML 1.0

Es un lenguaje para la descripción de mundos virtuales estáticos, que cumple tres requisitos fundamentales: es independiente de la plataforma donde se ejecute el visualizador, tiene capacidad para trabajar con un modelo eficiente en sistemas con conexiones lentas, y es extensible, es decir, susceptible de ser ampliado fácilmente [10].

### 2.- VRML 2.0

Después de la definición de la primera versión, se observó que los mundos estáticos no eran suficientes, sino que hacía falta que los objetos tuviesen comportamientos propios y que el usuario pudiese interactuar con ellos. Por ello, en 1995 el VAG solicitó propuestas de modificaciones de la especificación VRML que permitiesen lograr dichos objetivos. En 1996, la propuesta denominada *Moving Worlds*, presentada por *Silicon Graphics*, fue ratificada por el VAG como la especificación oficial VRML 2.0. Esta nueva versión es mucho más compleja que su predecesora, y en ella destacan los siguientes aspectos:



- Posibilidad de especificar comportamientos para los objetos, incluso usando el propio lenguaje VRML, mediante *scripts* en lenguajes externos (JavaScript, Java, Visual Basic, etc.), los cuales no están limitados por la especificación.
- Posibilidad de interacción con el usuario mediante la definición de una serie de sensores de posición, de contacto, de colisión, etc. La información registrada por estos sensores es enviada a los diferentes objetos que componen el mundo virtual y, en función de los valores recibidos, cada objeto virtual actuará en consecuencia.
- Finalmente, el lenguaje de descripción de escenas tridimensionales ha sido ampliado significativamente, posibilitando efectos de fondo, sonidos tridimensionales, nieblas, etc.

VRML 2.0 permite interactuar con el mundo virtual. Sin embargo, no es posible interactuar con otras personas que están accediendo al mismo mundo en el mismo instante. *Living Worlds* es la nueva propuesta de *Silicon Graphics* que está actualmente examinando el *VRML Consortium* para la nueva versión 3.0. Según esta propuesta, todas las personas que acceden a un mundo virtual pueden definir su propia representación en el entorno tridimensional, tanto para la detección de colisiones, como para el aspecto que el usuario tendrá ante el resto de visitantes [10].

## 5.2.- Herramientas necesarias para trabajar con VRML

Para construir y visualizar en VRML se necesitan dos herramientas básicas:

### 1.- Un visualizador o navegador

Se trata de un software que puede funcionar como un programa independiente o ser un *plug-in* (programa añadido) de un navegador de Internet como Netscape Navigator o Microsoft Internet Explorer.

Todos ellos tienen al menos tres características en común:

- Visualizan presentaciones tridimensionales de un archivo VRML.
- Ofrecen algún mecanismo para cambiar el punto de vista del usuario dentro del mundo virtual.
- Tienen algún medio para acceder a Internet.

Los visualizadores más conocidos son:

- Cosmo Player.

- Cortona (utilizado en este proyecto).

## 2.- Un editor

El editor permite crear objetos y mundos virtuales. Se puede utilizar un editor de texto y guardar el archivo con extensión *.wrl* para que lo reconozca el visualizador, o utilizar algún programa específico para VRML.

## 5.3.- Metodología de trabajo con VRML

Un proyecto VRML, de cualquier envergadura, puede ser dividido en las siguientes etapas:

### 1.- Especificación

En este paso se procede a realizar la descripción del proyecto, puntualizar el perfil del usuario, indicar los recursos necesarios y establecer el modo de interacción con el usuario. Es decir, en función del problema definido, se establece con precisión qué se propone lograr con el entorno virtual de enseñanza-aprendizaje.

### 2.- Planificación

La complejidad del proceso de elaboración de los materiales depende de la calidad pedagógica y académica deseada para el producto final, y de la variedad, alcance y complejidad en la integración de los medios empleados. En esta etapa se decide cuándo y cómo construir.

### 3.- Muestreo

En esta etapa se recaban los antecedentes acerca del objeto a modelar, es decir, toda la información necesaria para un adecuado modelado de los objetos que poblarán el mundo virtual a construir. Esta etapa incluye a su vez:

- **Elaboración de contenidos:** Se seleccionan los contenidos y se recopila información específica.
- **Preproducción del material:** En este paso se procede a darle al material su estructura definitiva, donde, de acuerdo a las características propias del medio utilizado, se presentarán los contenidos y actividades.
- **Supervisión:** En esta etapa se lleva a cabo una nueva revisión de lo realizado. El objetivo fundamental de este paso es comprobar que no se hayan producido distorsiones en los contenidos.

#### 4.- Diseño

Una vez conseguidos todos los antecedentes acerca de los diversos objetos a modelar, por medio del proceso de muestreo, se procede al diseño del mundo virtual. Esta etapa incluye a su vez las siguientes etapas:

- a) **Introducción al diseño:** Antes de iniciar el proceso de diseño, se revisa la sintaxis y estructura básica de un archivo con extensión “.wrl”.
- b) **Pasos preliminares en el diseño:** Esta etapa abarca los conceptos de:
  - **Identificación de objetos:** Para ello, se sigue un estudio similar al análisis orientado a objetos, identificando los mismos a partir de la evaluación de las herramientas que se utilizarán y los antecedentes obtenidos a partir del muestreo realizado. También se definen los sucesos más significativos para los componentes de cada sistema.
  - **Especificación de atributos:** En este caso, se entiende por atributos aquellos datos que le otorgan cualidades físicas relevantes a un objeto, para su modelado posterior. Con los objetos ya identificados, se procede a realizar una lista adecuada para la finalización del proyecto a realizar.
  - **Identificación de eventos:** Un evento involucra cambios en los valores de uno o más de sus atributos. Algunos de los cambios pueden ser de posición, color, tamaño, sonido, entre otros.
  - **Comunicación entre objetos:** Los objetos del mundo virtual se comunicarán, por medio de mensajes, quiénes deben ir encaminados por una ruta. Se identifica la existencia de estas interacciones, teniendo en cuenta la información obtenida y, además, se indicarán objetos que presenten variaciones en su movimiento o características físicas.

#### 5.- Construcción

Existen muchos métodos usados para crear VRML. Cada compañía tiene su propio “cocktail” de herramientas. Para el presente proyecto, las escenas se realizaron en el programa de diseño gráfico *Solid Edge* (un programa de modelado comúnmente utilizado para crear geometrías en 3D), guardando los archivos con extensión .wrl, que son los que se pueden exportar al entorno VRML. El archivo .wrl obtenido, se manipuló con el editor “WordPad” que permite operar, editar e imprimir archivos fuente. En esta etapa se obtuvo el modelo virtual descrito en la etapa de especificación.

## 6.- Pruebas

Debido a que VRML no es un lenguaje de programación (no se compila antes de lanzarlo) cualquier detección de errores en la sintaxis de los archivos se conoce en el momento en que estos son cargados en memoria. Esto permite conocer los posibles errores cometidos antes de cargar la simulación, lo que disminuye el tiempo transcurrido entre el momento en el que se detecta el problema, y el momento en el que se soluciona.

## 7.- Publicación

En esta etapa simplemente se coloca el archivo en un servidor *Web*, si fuera oportuno.

### 5.4.- Programación en VRML

A continuación, se describe la normativa, estructuras, comandos y procesos para la creación de los mundos virtuales en el lenguaje VRML.

#### 5.4.1.- Normas básicas para crear un mundo virtual

Las normas básicas que se deben tener en cuenta a la hora de crear un mundo en VRML son las siguientes:

- VRML es un lenguaje sensible a mayúsculas y minúsculas, lo cual ha de ser tenido en cuenta a la hora de asignar nombres.
- Todos los nodos han de comenzar siempre con letra mayúscula.
- Los campos de nodos deben comenzar siempre con letra minúscula.
- Los números se escriben en punto flotante.
- Utilizar una línea distinta para cada nodo, para cada campo y para cada valor de cada campo.
- Situar cada línea, según su jerarquía.
- Colocar cada símbolo de cierre en el nivel de jerarquía que le corresponde.
- Poner las líneas de comentario necesarias al mismo nivel que lo que se comenta.
- Poner nombres propios a los nodos.

### 5.4.2.- Estructura básica de VRML

Como se ha mencionado anteriormente, VRML es un lenguaje de descripción de escenas en el que cada escena se compone de un número de objetos. Los objetos pueden ser formas sólidas situadas y orientadas de determinada forma o elementos intangibles que afectan a la escena como luces, sonido y distintos puntos de vista. Para crear estos mundos de realidad virtual se utilizan ficheros de texto, cuya extensión es siempre *.wrl*, los cuales pueden ser desarrollados mediante cualquier editor o procesador de textos. Además, existe la posibilidad de utilizar programas de diseño gráfico, los cuales generan automáticamente ficheros en formato VRML [17].

Todo VRML está compuesto por los siguientes elementos:

- **Cabecera:** La cabecera de todo fichero VRML es siempre la misma:

**#VRML V2.0 utf8**

donde “VRML V2.0” indica el estándar empleado y “utf8” autoriza el uso de caracteres internacionales.

- **Comentarios:** En VRML un comentario se escribe en una sola línea, la cual comienza con el símbolo “#”. Se pueden tener tantas líneas de comentarios como se desee.
- **Nodos:** Un nodo es la estructura mínima indivisible de un fichero VRML y tiene como misión definir las características de un objeto, o bien, las relaciones entre distintos objetos. La mayoría de los nodos pueden repetirse tantas veces como sea necesario en una escena, salvo una serie de nodos especiales como los que definen la niebla o la panorámica del mundo virtual, que aparecen una sola vez.

Por otra parte, no todos los nodos afectan al aspecto visual del mundo. Por ejemplo, existen nodos que actúan como sensores que detectan acciones del usuario e informan de ellas a otros objetos, y otros que se encargan de modelar los sonidos.

Los nodos a su vez, contienen campos que describen propiedades. Todo campo tiene un tipo determinado y no se puede inicializar con valores de otro tipo. De este modo, cada tipo de nodo tiene una serie de valores predeterminados para todos sus campos, de forma que cuando se utilizan en una escena, sólo se deben indicar aquellos campos que se quieran modificar [17].

Los campos pueden ser simples o campos que indiquen a vectores u otros nodos. En la tabla 2 se muestran todos los campos con una descripción de cada uno de ellos.

<b>SFBool</b>	Es un valor booleano simple que puede tomar el valor “TRUE” o “FALSE”
<b>SFColor &amp; MFFloat</b>	Este campo contiene un color simple, conformado por tres números de punto flotante entre 0 y 1 que corresponden a los valores rojos, verdes y azules de ese color.
<b>SFFloat &amp; MFFloat</b>	SFFloat es un valor simple de punto flotante. MFFloat son varios valores de punto flotante. Puede tomar el valor de “TRUE” o “FALSE”
<b>SFImage</b>	SFImage es una imagen bidimensional
<b>SFInt32 &amp; MFInt32</b>	Uno solo o una lista de números enteros de 32 bits. Éstas pueden estar en forma decimal o hexadecimal. El valor se define ente $(+\infty, -\infty)$
<b>SFNode &amp; MFNode</b>	SFNode es un solo nodo, y MFNode es una lista de nodos.
<b>SFRotation &amp; MFRotation</b>	Estos campos especifican una rotación sobre un eje. Esto es realizado por cuatro números de punto flotante (X Y Z $\alpha$ )
<b>SFString &amp; MFString</b>	Contiene una lista de caracteres en el set de caracteres utf-8, debiendo estar entre comillas
<b>SFTime &amp; MFTime</b>	Un valor o una lista de valores de tiempo.
<b>SFVec2f &amp; MFVec2f</b>	Uno solo o una lista de vectores 2D (XY).
<b>SFVec3f &amp;</b>	Uno solo o una lista de vectores 3D (X Y Z).

**Tabla 2 – Campos de VRML**

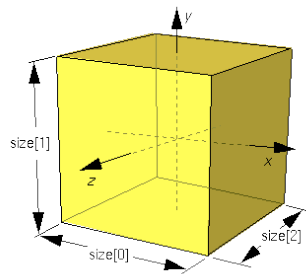
### 5.4.3. Nodos y comandos

Puesto que el objetivo de este proyecto no es realizar un tutorial sobre la programación en VRML sólo se describirán los nodos y los comandos utilizados para realizar este proyecto.

#### 5.4.3.1 Primitivas

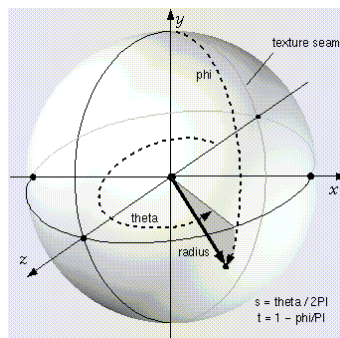
En VRML existe una serie de objetos predefinidos que forman el conjunto de primitivas de geometría. Estas primitivas son:

- **Box:** Para crear paralelepípedos genéricos como pueden ser cubos y cajas (Fig. 75).



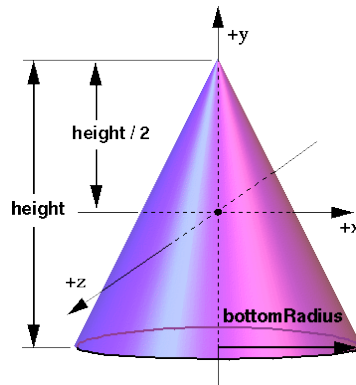
**Fig. 75 – Primitiva Box [16]**

- **Sphere:** Para crear esferas (Fig. 76).



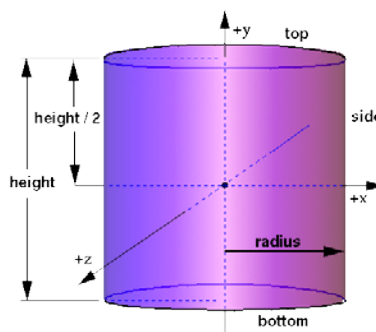
**Fig. 76 – Primitiva Sphere [16]**

- **Cone:** Para crear conos (Fig. 77).



**Fig. 77 – Primitiva *Cone* [16]**

- **Cylinder:** Para crear cilindros (Fig. 78).



**Fig. 78 – Primitiva *Cylinder* [16]**

Cada una de estas primitivas es un nodo.

#### 5.4.3.2. Node *SHAPE*

Para poder visualizar las primitivas, es necesario utilizar el nodo *Shape*. Este nodo tiene dos campos: *geometry* y *appearance*.

El campo *geometry* es donde se especifica la geometría de algún objeto 3D. Es a través de este campo que se especifican las primitivas. Estas primitivas son nodos. A continuación, se muestra cómo definir cada una de las primitivas existentes:



- **Box:** Se define una caja de X unidades de ancho, Y de alto y Z de fondo. El objeto queda definido de forma que su centro coincide con el origen de coordenadas locales. Se define como se indica a continuación:

```
Shape {  
    Geometry Box size {X Y Z}  
}
```

- **Sphere:** Se define una esfera de X unidades de radio. También queda centrada en el origen de coordenadas. Su código es el siguiente:

```
Shape {  
    Geometry Sphere {radius X}  
}
```

- **Cone:** Se define un cono con una base de X unidades de radio y una altura de Y unidades. También queda centrado en el origen de coordenadas. Así se define:

```
Shape {  
    Geometry Cone {  
        BottomRadius X  
        Height Y  
    }  
}
```

- **Cylinder:** Se define un cilindro con una base de X unidades de radio y una altura de Y unidades. También queda centrado en el origen de coordenadas. Su código es el siguiente:

```
Shape {  
    Geometry Cylinder {  
        Radius X  
        Height Y  
    }  
}
```

Para definir una apariencia de las primitivas (y en general de todos los objetos) se debe definir un color o una textura a aplicar a su superficie. A continuación, se explica cómo definir un color.

El *node Shape* dispone de un campo llamando *appearance* donde se definen las características del material del objeto. Para hacerlo, se utiliza el *node Appearance* de VRML.

Este nodo tiene diversos campos entre los que se encuentra *material*. Para definir este campo se debe utilizar el *node Material*, que a su vez posee un campo llamado *diffuseColor*, que hace referencia al color que refleja el objeto, es decir, el color del objeto.

Los colores en VRML se definen en RGB (*Red-Green-Blue* o Rojo-Verde-Azul). Los valores de R, G y B deben estar dentro del rango [0,1].

#### 5.4.3.3 Node GROUP

El *node Group* permite unir un conjunto de nodos de forma que actúen como una entidad única, pero sin efectuar ninguna transformación en ellos. La principal características de este tipo de grupo es que los objetos son todos creados en el mismo punto (en el centro del escenario de realidad virtual). El campo *children* dentro del nodo *Group* puede contener otros nodos. De este modo, los nodos pueden anidarse entre sí, creando una jerarquía de nodos. A continuación se muestra un ejemplo de cómo se puede definir este nodo:

```
Group {  
  
    eventIn      MFNode addChildren  
  
    eventIn      MFNode removeChildren  
  
    exposedField MFNode children []  
  
    field        SFVec3f bboxCenter 000 # (-∞,∞)  
  
    field        SFVec3f bboxSize -1 -1 -1 # (0, ) ó -1,-1,-1  
  
}
```

#### 5.4.3.4 Node *INLINE*

Hay veces que interesa incluir algún mundo ya creado dentro del que se está construyendo. El *node Inline* permite hacer exactamente esto, “llamando” a ese mundo desde su ubicación en el ordenador ó “URL”.

Se puede poner más de una URL, separadas por comas agrupadas dentro de corchetes []. Con esto, se define una lista en orden decreciente de preferencia. Por ejemplo, si el *browser* no puede acceder al archivo de la primera URL, entonces intenta acceder a la segunda, etc.

De la siguiente forma se programa VRML para hacer que aparezca alguno de los mundos ya creados de los que se hablaba antes (en el caso de este proyecto, las piezas generadas por *Solid Edge*):

```
Inline {  
  
    exposedField MFString url []  
  
    field        SFVec3f bboxCenter 000 # (-∞,∞)  
  
    field        SFVec3f bboxSize -1 -1 -1 # (0, ) ó -1,-1,-1  
  
}
```

#### 5.4.3.5 Node *TRANSFORM*

Para situar, orientar y escalar los objetos según las necesidades dentro del entorno que se esté construyendo, se deben poder aplicar las transformaciones geométricas básicas, que son: traslación, rotación y escalado. También existen variantes de estas pero no se verán aquí.

Para aplicar transformaciones en VRML se utiliza el node Transform. Este nodo puede realizar cualquier de las tres operaciones o las tres a la vez.

```
Transform {  
    eventIn      MFNode    addChildren  
    eventIn      MFNode    removeChildren  
    exposedField SFVec3f    center 000 # (-∞,∞)  
    exposedField MFNode     children []  
    exposedField SFRotation  rotation 0010# [-1,1] ,(-∞,∞)  
    exposedField SFVec3f     scale 111# (0,∞)  
    exposedField SFRotation  scaleOrientation 0010# [-1,1] ,(-∞,∞)  
    exposedField SFVec3f     translation 000 # (-∞,∞)  
    field        SFVec3f     bboxCenter 000 # (-∞,∞)  
    field        SFVec3f     bboxSize -1-1-1# (0,∞) o -1,-1,-1  
}
```

El campo *translation* es donde se define el desplazamiento que sufre el objeto que se desea trasladar. Así pues, se define un desplazamiento a lo largo de los ejes (X Y Z). Las traslaciones son relativas y, por lo tanto, se suman a la posición actual de los objetos.

El campo *rotation* es donde se define la rotación que sufre el objeto a modificar. Este campo tiene en realidad dos partes. Los tres primeros valores determinan un vector en el espacio 3D que corresponde a la orientación del eje

de rotación de la transformación. El eje de rotación coincide con los ejes coordenados cuando:

- Eje rotación X: Se define de la siguiente forma: (1 0 0).
- Eje rotación Y: Se define de la siguiente forma: (0 1 0).
- Eje rotación Z: Se define de la siguiente forma: (0 0 1).

Este sistema de definir un eje de rotación de la transformación es muy flexible y genérico, pero resulta compleja cuando el eje de rotación no es paralelo a un eje coordenado.

La segunda parte es el ángulo de la rotación que se quiere aplicar (expresado en radianes) respecto al eje definido en la primera parte. Al estar en un sistema de ejes a derechas, los ángulos positivos resultan en rotaciones en sentido anti horario (y los ángulos negativos en sentido horario).

El campo *scale* es donde se define el escalado no uniforme que sufre el objeto que se desea modificar. Para definir el escalado, se indica el tanto por uno que se quiere agrandar/reducir el objeto en cada una de las direcciones que marcan los ejes coordenados. Si se desea aplicar un escalado uniforme, solo es necesario poner el mismo valor en las tres direcciones.

#### 5.4.3.6 Sensor de tiempo

En VRML97 se dispone de una herramienta muy potente, el sensor de tiempo (*TimeSensor*). Este es un reloj que se puede utilizar para aprovechar el paso del tiempo como motor para mover objetos, cambiar su color, variar orientaciones, etc. Este se basa en el reloj real del sistema, midiendo el tiempo actual en segundos transcurridos a partir de las 00:00 horas del 1º de enero de 1970.

Los campos más característicos son el *cycleInterval*, en el que se pone el ciclo de tiempo elegido para mover los objetos, y *loop*, que sirve para que ese ciclo de tiempo elegido se repita una vez termine, si procede.

La programación del *TimeSensor* se indica a continuación:

```
TimeSensor {  
    exposedField    SFTime    cycleInterval 1    #(0,∞)  
    exposedField    SFBool    enabled            TRUE  
    exposedField    SFBool    loop              FALSE  
    exposedField    SFTime    startTime          0    #(-∞,∞)  
    exposedField    SFTime    stopTime           0    #(-∞,∞)  
    eventOut        SFTime    cycleTime  
    eventOut        SFFloat    fraction_changed 0    #[0,1]  
    eventOut        SFBool    isActive  
    eventOut        SFTime    time  
}
```

#### 5.4.3.7. *Sensor de tacto*

El sensor de tacto nos permite detectar cuándo el observador apunta a un objeto con el cursor y cuándo oprime el botón del ratón. Por lo tanto permite definir “botones 3D”.

El sensor de tacto ha de ser agrupado junto con los objetos que sirven de botón, dentro de algún nodo de agrupación, para que VRML sepa qué objetos están activos cuando el usuario aprieta el botón.

La programación del *TouchSensor* se indica a continuación:

```
TouchSensor {  
  
    exposedField    SFBool enable TRUE  
  
    eventOut        SFVec3 hitNormal_changed  
  
    eventOut        SFVec3 hitPoint_changed  
  
    eventOut        SFVec2 hitTexCoord_changed  
  
    eventOut        SFBool isActive  
  
    eventOut        SFBool isOver  
  
    eventOut        SFTIME touchTime  
  
}
```

#### 5.4.3.8 Interpoladores

La interpolación lineal (que es la que utiliza VRML97) es un concepto matemático que permite definir dos puntos (en cualquier dimensión) y calcular un punto intermedio sobre la recta que los une a partir de especificar el tanto por ciento del recorrido entre los dos puntos.

Los dos campos principales son *key* y *keyvalue*. El primero indica los estados por los que va pasar el elemento y el segundo el valor que va a alcanzar la variable en cada uno de estos estados.

A continuación, se muestra cómo definir cada uno de los interpoladores:

- *ColorInterpolator*: Interpola colores. El código para su programación es el siguiente:

```
ColorInterpolator {  
  
    eventIn          SFFloat set_fraction    #(-∞,∞)  
  
    exposedField      MMFloat key             [] #(-∞,∞)  
  
    exposedField      MMColor keyValue       [] #(0,1)  
  
    eventOut          SFColor value_changed  
  
}
```

- *CoordinateInterpolator*: Interpola coordenadas de vértices. El código para su programación es el siguiente:

```
CoordinateInterpolator {
    eventIn      SFFloat set_fraction      #(-∞,∞)
    exposedField MFFloat key                []    #(-∞,∞)
    exposedField MFVec3f keyvalue []        #(-∞,∞)
    eventOut     MFVec3f value_changed

}
```

- *NormalInterpolator*: Interpola normales a superficies. El código para su programación es el siguiente:

```
NormalInterpolator {
    eventIn      SFFloat set_fraction      #(-∞,∞)
    exposedField MFFloat key                []    #(-∞,∞)
    exposedField MFVec3f keyvalue []        #(-∞,∞)
    eventOut     MFVec3f value_changed

}
```

- *OrientationInterpolator*: Interpola ángulos de rotación. El código para su programación es el siguiente:

```
OrientationInterpolator {
    eventIn      SFFloat set_fraction      #(-∞,∞)
    exposedField MFFloat key                []    #(-∞,∞)
    exposedField MFRotation keyValue []      #[-1,1],(-∞,∞)
    eventOut     SFRotation value_changed

}
```



- *PositionInterpolator*: Interpola posición de objetos. El código para su programación es el siguiente:

```
PositionInterpolator {  
    eventIn      SFFloat set_fraction      #(-∞,∞)  
    exposedField MFFloat key                #(-∞,∞)  
    exposedField MFVec3f key value []       #(-∞,∞)  
    eventOut      SFFloat value_changed  
}
```

- *ScalarInterpolator*: Interpola valores cualquiera (escalares, es decir, univaluados). El código para su programación es el siguiente:

```
ScalarInterpolator {  
    eventIn      SFFloat set_fraction      #(-∞,∞)  
    exposedField MFFloat key                #(-∞,∞)  
    exposedField MFVec3f key value []       #(-∞,∞)  
    eventOut      SFFloat value_changed  
}
```

#### 5.4.3.9. Puntos de vista

En VRML 97 existe la posibilidad de definir puntos de vista diversos. De esta forma, se pueden mostrar al usuario aquellas partes que son más importantes del mundo que se ha definido. El usuario puede navegar con total libertad por el mundo creado. Pero si se define una serie de puntos de interés, puede ir directamente a estos. La forma de definir estos puntos de vista o puntos de interés es mediante el *node Viewpoint*.

Para escoger cualquier punto de vista, se utiliza el comando que el *browser* normalmente tiene disponible. El primer punto de vista que se define en el código, pasa a ser el punto de vista de arranque del mundo.

El primer *node Viewpoint* define un punto de vista idéntico al que ya se establece por defecto al entrar al *browser*. Esto sirve para poder volver a la posición inicial sin tener que volver a cargar el entorno. Los valores por defecto son pues, situar el observador sobre el eje Z positivo a 10 unidades del origen y mirando en dirección del eje Z negativo.

Este es el código para la programación del nodo *Viewpoint*:

```
Viewpoint {  
    eventIn      SFBool      set_bin  
    exposedField SFFloat      fieldOfView 0.785398 # (0, $\pi$ )  
    exposedField SFBool      jump      TRUE  
    exposedField SFRotation   orientation 0 0 1 0 #[-1,1],(- $\infty$ , $\infty$ )  
    exposedField SFVec3f      position   0 0 1 0 #(- $\infty$ , $\infty$ )  
    field        SFString     description ""  
    eventOut     SFTIME       bindTime  
    eventOut     SFBool       isBound  
}
```

#### 5.4.3.10 Node *SCRIPT*

Aunque en VRML existen muchos mecanismos para definir interacción y comportamientos, hay cosas que no se pueden hacer directamente y entonces, se debe utilizar la potencia de un lenguaje de programación externo. Esto se consigue a través del *node Script* y los lenguajes que se pueden utilizar son el Java y el JavaScript (o compatible con JavaScript).

En este proyecto, tan sólo se ve la utilización de JavaScript dentro del *node Script* debido a que es mucho más sencillo, directo y común de utilizar.

El *node Script* permite los siguientes pasos:

- Captar los eventos que se necesiten.
- Conseguir los valores que forman los eventos y procesarlos.
- Enviar los resultados de este proceso como eventos de salida (para ser encaminados, mediante *ROUTEs*, hacia otros nodos).

Un *node Script* está formado por dos partes principales: las definiciones de campos y eventos, y el código en el lenguaje que se ha escogido.

Es importante el hecho de que este nodo pueda definir campos y eventos según las necesidades, en contraste con los otros nodos de VRML que ya tienen predefinidos todos sus componentes.

El nodo *Script* queda definido en VRML de la siguiente forma:

```
Script {  
  
    exposedField  MFString url []  
  
    field         SFBool directOutput FALSE  
  
    field         SFBool mustEvaluate FALSE  
  
    eventIn       eventType eventName  
  
    field         fieldType fieldName initialValue  
  
}
```

#### 5.4.3.11 Eventos

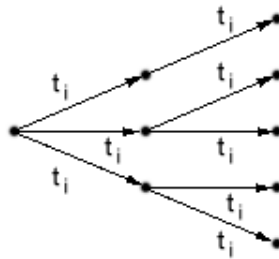
Un evento es como un mensaje que puede ser enviado por un objeto y capturado por otro. Puede haber eventos indicando que un nodo ha cambiado de posición, o que ha pasado de un estado inactivo a uno activo, que ha pasado una fracción de tiempo, etc.

En teoría, los *browsers* tienen un reloj interno que va marcando en tiempo real el paso del tiempo. Los eventos se van produciendo y recibiendo a cada intervalo de tiempo de reloj (en teoría de forma continua), pero en la práctica, normalmente se puede asumir que los *browsers* marcan el paso del tiempo a cada *frame* (cuadro o imagen) generado gráficamente. Así pues, cada evento tiene una especie de sello de tiempo, del instante en que se ha generado (emitido, enviado).

Evidentemente, un evento es emitido por un *eventOut* y debe ser recibido por un *eventIn*. Pero también puede ocurrir que un evento sea emitido o recibido por un *exposedField*, que es un campo de entrada y salida.

Como es de suponer, tanto el evento que envía como el que recibe, han de ser del mismo tipo. Por lo tanto, un evento enviado por un *eventOut* de tipo *SFInt32* (entero 32 bits univaluado) sólo puede ser recibido por un *eventIn* (o *exposedField*) de tipo *SFInt32*.

Puede ocurrir que un objeto (nodo) reciba un evento, el cual, le fuerce a enviar uno a él mismo. En este caso, se genera lo que se conoce por cascada de eventos. En una cascada de eventos (Fig. 79), todos los eventos implicados tienen el mismo sello de tiempo.



**Fig. 79 – Cascada de eventos.**

#### 5.4.3.12 ROUTE, DEF Y USE

Cuando se quiere definir un objeto que se utiliza en repetidas ocasiones y se satura el código por el hecho de repetir un mismo bloque de código, entonces se puede utilizar *DEF* y *USE*. Estos dos comandos permiten dar un nombre a un cierto nodo que, posteriormente, se utilizará con la misma estructura y definición en diversos sitios.

El comando *ROUTE* se utiliza para hacer que los campos de los nodos definidos en la creación de los elementos que forman el mundo virtual que requieran cambiar, puedan hacerlo a los mismos campos que se han definido para otros elementos de forma diferente. Por ejemplo, se puede definir el color de un elemento que en un principio se requiere rojo, para más tarde, hacer que cambie al color de otro elemento que se definió azul.

A continuación, se muestra cómo se debe programar en VRML para la utilización de los comandos *DEF* y *USE*:

```
DEF MiRojo Sppearance {  
    Material Material {  
        diffuseColor 0.8 0.2 0.4  
    }  
}  
  
Transform {  
    Translation -2 0 0  
    Children [  
        Shape {  
            Geometry Box {size 3 3 3}  
            Appearance USE MiRojo  
        }  
    ]  
}  
  
Transform {  
    Translation 2 0 0  
    Children [  
        Shape {  
            Geometry Sphere {radius 2}  
            Appearance USE MiRojo  
        }  
    ]  
}
```

En el fragmento de código anterior, se observa cómo se ha definido con *DEF* una apariencia que después se usa con *USE* en dos objetos diferentes. Lo que se ha hecho en realidad es sencillamente darle el nombre *MiRojo* a un trozo de código para después ahorrar el escribirlo muchas veces.

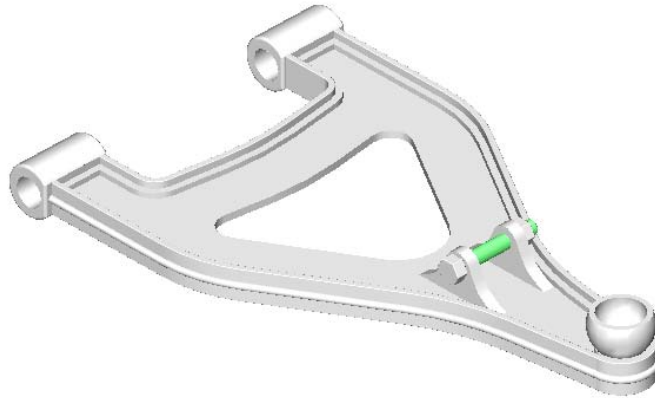
Más importante que el ahorro en escritura, es la gran ayuda que supone cuando se decide variar la definición de color de *MiRojo*, ya que si se tiene la definición de color repetida en distintos lugares, es necesario encontrar todas las repeticiones y modificar cada una de ellas. En cambio, si está definida únicamente al principio, tan sólo se tiene que variar esta para que, automáticamente, se modifique en todos los lugares donde ha sido usado.

## **CAPITULO 6.- CREACION DE LA SUSPENSION EN EL MUNDO VIRTUAL**

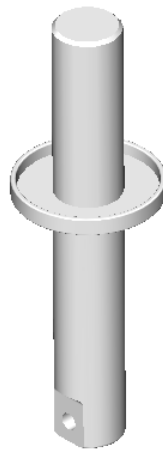
En este capítulo se describe de qué forma se ha realizado el ensamblado de cada una de las piezas que componen la suspensión doble *Wishbone*, objeto de este proyecto, en el mundo virtual, así como el proceso de simulación del movimiento de la misma.

Para hacer más sencillo su ensamblaje y animación, en primer lugar se han separado las piezas por grupos entre los que no existe movimiento relativo entre sí. En la mayoría de los casos, estos conjuntos son las piezas sueltas. Excepto en el caso de la sujeción base y el amortiguador, donde no existe movimiento relativo entre estas piezas, y los tornillos y las tuercas que las componen.

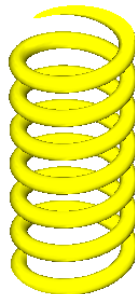
Las figuras 80 a 85, que se muestran a continuación, son el resultado de pasar los elementos que componen la suspensión doble *Wishbone*, de *Solid Edge* al lenguaje VRML.



**Fig. 80 – Sujeción base**

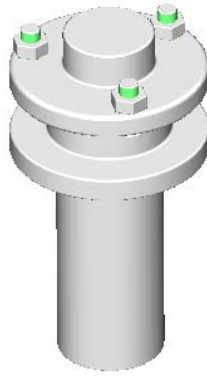


**Fig. 81 – Sujeción amortiguador**

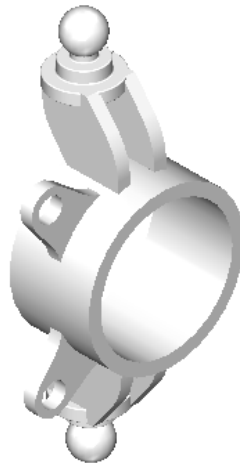


**Fig. 82 – Muelle**

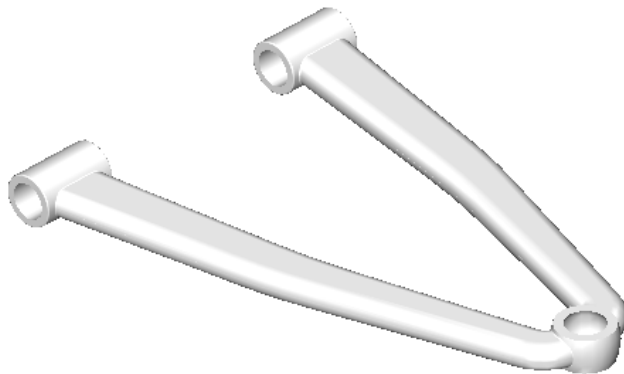




**Fig. 83 – Amortiguador**



**Fig. 84 – Mangueta**



**Fig. 85 – Sujeción superior**

## 6.1. Introducción de las piezas en el mundo virtual

Antes de introducir las piezas, se crea el espacio virtual en el que se trabajará. Todo el código que se debe poner, previo a la definición de las piezas y que crea el mundo virtual, lo proporcionan por defecto los archivos *.wrl* que se obtienen directamente desde *Solid Edge*, y que se puede ver a continuación:

```
#VRML V2.0 utf8 Solid Edge VRML Export V1.2
```

```
Group {  
  children [  
    WorldInfo { title "Produced using Solid Edge VRML Export, by UGS PLM  
Solutions." }  
    DEF Main Viewpoint {  
      position 23.8 -23.8 23.9  
      orientation 0.71 0.304 0.554 1.25  
      fieldOfView      0.00793  
      description "Main View"  
    }  
    DEF Top Viewpoint {  
      position 0 0 12.4  
      orientation 1 0 0 0  
      fieldOfView      0.00793  
      description "Top View"  
    }  
    DEF Bottom Viewpoint {  
      position 0 0 -12.1  
      orientation 1 0 0 3.14
```

```
fieldOfView      0.00793
description "Bottom View"
}

DEF Front Viewpoint {
  position 0 -38.1 0.137
  orientation 1 0 0 1.57
  fieldOfView      0.00793
  description "Front View"
}

DEF Back Viewpoint {
  position 0 38.1 0.137
  orientation 0 0.707 0.707 3.14
  fieldOfView      0.00793
  description "Back View"
}

DEF Right Viewpoint {
  position 38.1 0 0.137
  orientation 0.577 0.577 0.577 2.09
  fieldOfView      0.00793
  description "Right View"
}

DEF Left Viewpoint {
  position -38.1 0 0.137
  orientation 0.577 -0.577 -0.577 2.09
  fieldOfView      0.00793
```

```
description "Left View"
}

DEF TopFrontRightIso Viewpoint {
  position 27.8 -27.8 14
  orientation 0.783 0.335 0.447 1.46
  fieldOfView 0.00793
  description "TopFrontRightIso View"
}

DEF TopFrontLeftIso Viewpoint {
  position -27.8 -27.8 14
  orientation 0.783 -0.335 -0.447 1.46
  fieldOfView 0.00793
  description "TopFrontLeftIso View"
}

DEF TopBackLeftIso Viewpoint {
  position -27.8 27.8 14
  orientation -0.234 0.547 0.781 3.75
  fieldOfView 0.00793
  description "TopBackLeftIso View"
}

DEF TopBackRightIso Viewpoint {
  position 27.8 27.8 14
  orientation 0.234 0.547 0.781 2.54
  fieldOfView 0.00793
  description "TopBackRightIso View"
```

```
}  
NavigationInfo {  
  type [ "EXAMINE","ANY" ]  
  headlight TRUE  
}  
Background { skyColor 1 1 1 }  
Group {  
  children [  
    DEF RightLight DirectionalLight {  
      direction -0.632455 0.547723 -0.547723  
      intensity 0.35  
    }  
    DEF LeftLight DirectionalLight {  
      direction 0.632455 0.547723 -0.547723  
      intensity 0.35  
    }  
    DEF BackLight DirectionalLight {  
      direction 0 -0.707107 -0.707107  
      intensity 0.35  
    }  
    DEF BottomLight DirectionalLight {  
      direction 0 0 1  
      intensity 0.35  
    }  
  ]  
}
```

```
}  
]  
}
```

A continuación, se deben definir los conjuntos que se mencionaban antes (Figs. 80 – 85). Para ello, se utiliza siempre la misma estructura. El comando *DEF*, que define la pieza, y el comando *Inline*, que “llama” a la pieza desde el archivo *.wrl* obtenido de *Solid Edge*. A continuación se detalla cómo se debe programar la definición de las piezas de la suspensión:

```
DEF Sujecion_base Transform {  
    translation 0.364 0 0.029  
  
    rotation 0 0 0 0  
  
    children[  
        Inline{  
            url"Sujecion_base_Mod.wrl"  
        }  
    ]  
}
```

Los ejes de coordenadas de todas las piezas en el mundo virtual coinciden con los ejes de coordenadas que se definieron en la creación de la pieza en *Solid Edge*.

A medida que se va introduciendo cada una de las piezas, se deben colocar según la posición que corresponda dentro del conjunto global de la suspensión. Como se puede observar en el fragmento de código anterior, para mover la pieza del origen de coordenadas, se usa el nodo *Transform*. Con él, se trasladan (*translation*), se rotan (*rotation*) o se escalan (*scale*) las piezas. En el caso de que no aparezca alguno de estos atributos, o su valor sea nulo (0 0 0 0), la pieza se quedará en la posición original.

Muchas veces se requiere que las piezas queden definidas dentro de otra que ya se definió previamente. Esto se hace para conseguir que la ubicación de la nueva pieza quede referenciada con respecto a la ubicación de la pieza ya definida, lo que facilita las cosas. Además, se debe tener en cuenta que cualquier movimiento o cambio que se defina para la pieza origen, automáticamente afectará a la pieza que queda definida dentro de la original.

A continuación, se muestra un fragmento de código en el que se ha usado este procedimiento:

```
DEF Sujecion_base Transform {  
    translation 0 0 0  
    rotation 0 0 0 0  
    children [  
        Inline {  
            url"Sujecion_base_Mod.wrl"  
        }  
        DEF Sujecion_amortiguador Transform {  
            translation -0.11 0 -0.01  
            rotation 0 1 0 -0.3  
            children [  
                Inline {  
                    url"Sujecion_amortiguador_Mod.wrl"  
                }  
            ]  
        }  
    ]  
}
```

Como se puede observar, se define “Sujecion\_amortiguador” dentro de “Sujecion\_base”. Así, “Sujecion\_amortiguador” se mueve, con respecto a “Sujecion\_base”, de la siguiente forma:

- “translation -0.11 0 -0.01”: Se traslada 0.11 unidades en la dirección negativa del eje “X”, nada en la dirección del eje “Y”, y 0.01 unidades

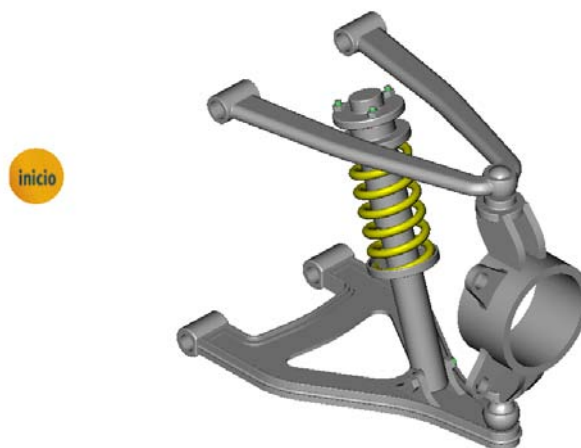
negativas en la dirección del eje “Z”, con respecto a la posición original de “Sujecion\_base”.

- “rotation 0 1 0 -0.3”: Se rota únicamente con respecto al eje “Y” [0 1 0], un ángulo de 0.3 radianes en sentido negativo, con respecto a la posición original de “Sujecion\_base”.

## 6.2. Movimientos de la suspensión

La simulación que se presenta en este proyecto, muestra el comportamiento de una suspensión doble *Wishbone* en un vehículo, en el momento en el que una de sus ruedas se encuentra con un resalto en su trayectoria.

Lo primero que aparece al abrir la presentación es el “mundo virtual VRML”, la suspensión objeto de nuestro estudio, y un botón de inicio (Fig. 86). En el anterior capítulo ya se explicó cómo se introducen las piezas de la suspensión. Para el botón de inicio, se crea directamente en VRML un cilindro con un pequeño espesor. Mediante el comando “texture ImageTexture { }” se le pone una imagen de “INICIO”, y mediante el comando *BILLBOARD*, se consigue que el botón siempre esté de cara al espectador, aunque este gire ó mueva libremente la pantalla de la presentación. Además, se define como *TouchSensor*, con lo que se consigue que hasta que no se pulse este botón no comience la simulación.

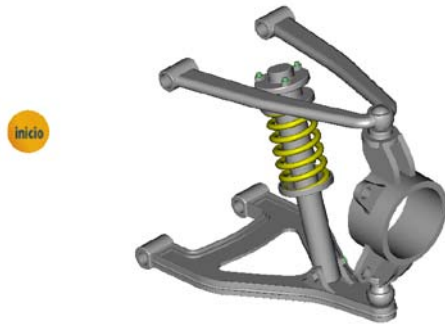


**Fig. 86 – Pantalla inicial de la simulación**

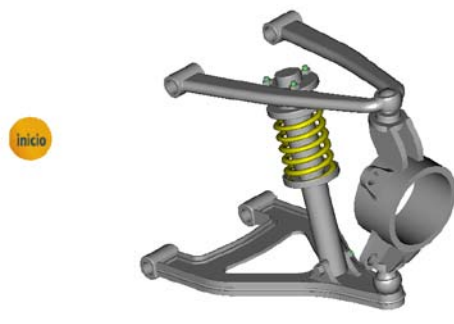
Para dar comienzo a la simulación, se debe pulsar el botón “INICIO”. El movimiento de todo el conjunto de la suspensión tiene tres fases:



- Un movimiento vertical de la mangueta de la suspensión hacia el chasis del vehículo, provocado por el impacto del resalto con la rueda (Fig. 87).
- Llegada al punto en el que el muelle ha absorbido toda la energía del impacto (Fig. 88).
- Relajación del muelle hasta el punto inicial de forma gradual y sin rebote debido a la acción del amortiguador (Fig. 87).



**Fig. 87 – Posición de compresión y relajación durante el recorrido del muelle**



**Fig. 88 – Posición de compresión máxima del muelle**

La duración del movimiento es de cinco segundos, según se definió en el *TimeSensor*.

Para definir el movimiento de cada una de las piezas, se usan los interpoladores. Con ellos, se define la posición de cada pieza en cada uno de los momentos que se han comentado anteriormente. En la mayoría de las piezas, los interpoladores se usan para hacer rotar las piezas. Es lo que ocurre en el caso de todas las sujeciones y la mangueta, la cual gira con respecto a la rótula inferior. Para la deformación del muelle, se ha utilizado un interpolador de posición. Esto se debe a que, a pesar de que lo que se modifica en el muelle es su escala, los atributos que se deben modificar en los interpoladores de posición son los mismos para realizar una traslación, que para escalar algún objeto dentro del mundo virtual.

La relación entre los movimientos y rotaciones definidos en los diferentes interpoladores, con cada una de las piezas a las que afecta, se consigue mediante el comando “ROUTE”. A través de este comando, los atributos que se quieren modificar de cada pieza, en este caso su movimiento, se relacionan con las posiciones que se han definido en los interpoladores.

Al final, lo que se hace realmente es definir tres posiciones para cada una de las piezas que componen la suspensión. Una vez pulsado el botón “INICIO”, las piezas comienzan a moverse hacia esas posiciones que se han definido en los interpoladores, creando así, de manera automática, el movimiento.

## **CAPITULO 7. CONCLUSIONES Y TRABAJOS FUTUROS**

### **7.1.- Conclusiones**

Las conclusiones obtenidas tras la realización del presente proyecto son:

- El entorno VRML proporciona una gran facilidad para la introducción de piezas en el mundo virtual, modeladas en 3D, mediante programas de diseño asistido por ordenador. Esto es una ventaja, ya que no son necesarias demasiadas herramientas para la creación de una simulación virtual. Lo único que se necesita es un programa de diseño 3D (*Solid Edge*), un editor de texto (Word, WordPad, etc.) y un visualizador o navegador de Internet.
- La programación en VRML es relativamente sencilla. Esto supone otra ventaja para la enseñanza y el estudio de cualquier materia a través de esta herramienta: no son necesarios unos estudios previos o una excesiva especialización en el mundo de la programación para poder crear estas simulaciones. Un proceso complicado en el aprendizaje de una herramienta puede suponer un rechazo por parte del estudiante.
- La simulación por ordenador en un entorno virtual facilita la comprensión acerca del funcionamiento de un mecanismo, ya que permite observar la misión y el lugar que ocupa cada una de las piezas que lo forman.

## 7.2.- Trabajos futuros

En la misma línea que este proyecto, a continuación se proponen una serie de posibles trabajos futuros:

- Añadir al presente proyecto componentes mecánicos que acerquen la simulación a un modelo más real: ruedas, chasis del vehículo, bujes, *silentblocks*, etc. Esto permitirá una mejor comprensión del comportamiento de la suspensión.
- Realizar la simulación para una irregularidad diferente, como un badén o resaltos con diferentes geometrías
- Simular el comportamiento de la suspensión bajo la acción de aceleraciones y frenadas, o bajo diferentes distribuciones de carga.
- Hacer simulaciones para diferentes tipos de suspensiones, con otro tipo de muelles, con diferentes rigideces; ballestas, etc.
- Realizar simulaciones similares para otros sistemas mecánicos, como el sistema de frenos, los embragues, etc. Esto puede servir tanto a profesores, complementando sus explicaciones, como a los alumnos, mejorando la comprensión del funcionamiento de cualquier mecanismo.

## **8.- BIBLIOGRAFIA**

[1] Juan Alberdi Urbieto. Manuales de Automoción *TECNUN*. *Amortiguadores y suspensión*. Campus tecnológico de la Universidad de Navarra. 2009.

[www.tecnun.es/automocion](http://www.tecnun.es/automocion)

[2] [www.mecanicavirtual.org](http://www.mecanicavirtual.org). 2009.

[www.mecanicavirtual.org](http://www.mecanicavirtual.org)

[3] <http://es.cars.yahoo.com>. 2010.

<http://es.cars.yahoo.com/consejos/mecanica/amortiguadores.html>.

[4] Del Pino Galisteo, Miguel Ángel. *Simulación virtual de un freno de disco en entorno VRML*. Proyecto fin de carrera. Dpto. de Ingeniería Mecánica. Universidad Carlos III de Madrid. 2007.

[5] Fernández Casas, Raúl. *Modelado de un sistema de suspensión de un vehículo automóvil mediante SIMMECHANICS*. Proyecto fin de carrera. Dpto. de Ingeniería Mecánica. Universidad Carlos III de Madrid. 2008.

[6] Josep Camós. *El sistema de suspensión, el gran olvidado de la seguridad activa*. 2009.

<http://www.circulaseguro.com/2008/03/30-el-sistema-de-suspension-el-gran-olvidado-de-la-seguridad-activa>

[7] Silvia Elstein. *Nuevas Tecnologías, nuevos entornos sociales y culturales*. Agencia N° 3 Universidad Nacional Río Cuarto, Córdoba (Argentina). 2009.

<http://www.unrc.edu.ar/publicar/cde/Elstein.htm>

[8] Introducción a *Solid Edge*. UGS Series. 2009.

[www.solidedge.com](http://www.solidedge.com)

[9] Unigraphics Solutions Inc. *Solid Edge. Manual del usuario*. Versión 10. 2001.

[10] José R. Hilera. *Aplicación de la Realidad Virtual en la enseñanza a través de Internet*. Departamento de Ciencias de la Computación. Universidad de Alcalá. 2009.

<http://www.ucm.es/info/multidoc/multidoc/revista/num8/hilera-oton.html>

[11] [www.activamente.com](http://www.activamente.com). 2010.

<http://www.activamente.com.mx/vrml/>

[12] Andrea L. Ames, David R. Nadeau, John L. Moreland. *VRML 2.0 Sourcebook*. Ediciones John Wiley & Sons. 1996.

[13] <http://rvrealidadvirtual.blogspot.com/>. 2010.

<http://rvrealidadvirtual.blogspot.com/>

[14] [www.cicese.mx](http://www.cicese.mx). 2010.

<http://telematica.cicese.mx/computo/super/cicese2000/realvirtual/Part4.html>

[15] Virtual Reality Laboratory. The State University of New York at Buffalo. 2010.

<http://wings.buffalo.edu/academic/departament/eng/mae/vrlab/>

[16] Narcís Parés. *Tutorial de VRML 97*. 2009.

[http://www.iua.upf.es/~npares/docencia/vrml/tutorial\\_e.htm](http://www.iua.upf.es/~npares/docencia/vrml/tutorial_e.htm)

[17] Carlos L. González Cordero. *Tutorial de VRML 2.0*. 2009.

<http://www.wdi.ujaen.es/~rsegura/igai/vrml/documentos/tema0.htm>

[18] Antonio Ramires Fernandes. *VRML Interactive tutorial*. 2009.

<http://www.lighthouse3d.com/vrml/tutorial/>

[19] <http://www.monografias.com>. 2010.

<http://www.monografias.com/trabajos2/vrml/vrml.shtml>